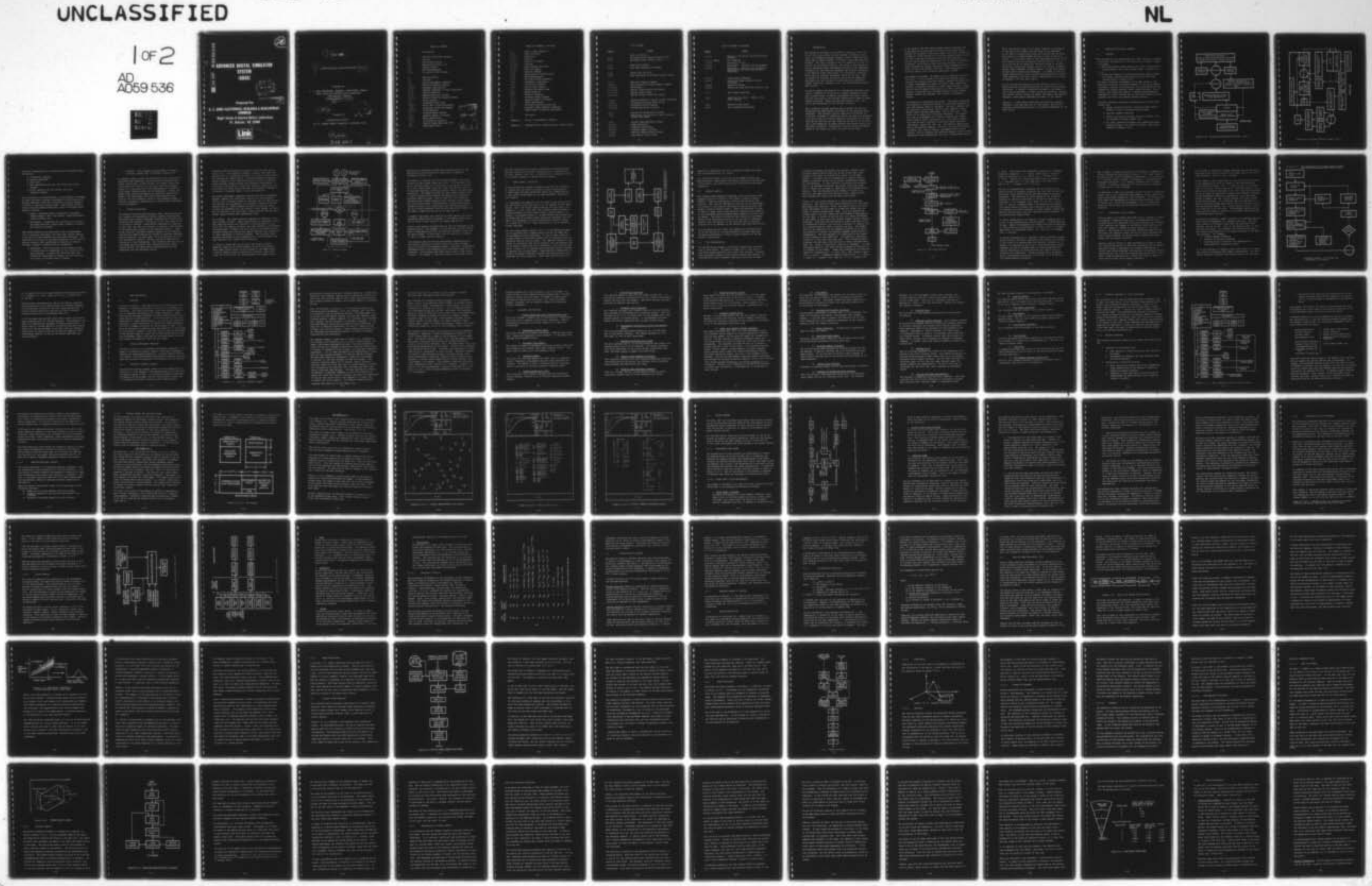
AD-A059 536 SINGER CO SUNNYVALE CA LINK DIV F/G 15/4
ADVANCED DIGITAL SIMULATOR SYSTEM (ADSS).(U)
JUL 78 DAAK70-77-C-0175
UNCLASSIFIED NL

1 OF 2
AD
A059 536



ADVANCED DIGITAL SIMULATOR SYSTEM (ADSS)

Prepared For

U. S. ARMY ELECTRONICS, RESEARCH & DEVELOPMENT COMMAND

Night Vision & Electro-Optics Laboratory

Ft. Belvoir, VA 22060

Link

A DIVISION OF THE SINGER COMPANY

9 FINAL REPORT.

6 ADVANCED DIGITAL SIMULATOR SYSTEM (ADSS).

Prepared For

U. S. ARMY ELECTRONICS, RESEARCH & DEVELOPMENT COMMAND
NIGHT VISION AND ELECTRO-OPTICS LABORATORY
FORT BELVOIR, VIRGINIA
CONTRACT DAAK70-77-C-0175

15

12 168p.

Prepared By

SINGER/LINK DIVISION
1077 E. ARQUES AVENUE, SUNNYVALE, CALIFORNIA 94086

11 JULY 1978

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

LIST OF FIGURES

LIST OF FIGURES (Continued)

1. INTRODUCTION

The purpose of the ADSS is to provide a vehicle by which the NVL can test and evaluate various classes of sensors, either singly or as complete weapons systems, without committing the large sums of money necessary to actually design and fabricate hardware. The latter course is not only expensive but may not produce the anticipated results. Far worse is the possibility that an expensive sensor development program may be completely successful but the resulting equipment not provide information to an operator either fast enough, or sufficiently interpretable that its tactical usefulness justifies its cost. What is required is a method by which a device can be completely simulated; not only its own intrinsic qualities (bandwidth, ranges, resolution, frequency, etc.) but its operating environment as well.

An entire industry has been built to accomplish realistic simulation. Starting at the point of a simple responsive device that trained pilots to observe instruments and handle a small aircraft the industry now provides complete multi-station weapon system simulation. These provide every sensor, aural, and motion cue encountered in any tactical situation.

Early simulator systems employed analog devices and as such, encountered the same system limitations of the equipment being simulated. More recently digital computing circuits and devices have increased in capability and at the same time decreased in price so that with digital computing techniques we can simulate devices that are not only state of the art but we can simulate a new generation of sensor devices that cannot actually be realized in practice. The ability to build digital environment models at any scale also falls into this latter class.

At the present time many large weapons system contracts have either been let or have been delivered that provide digital systems for microwave, electronic sensor, and visual simulation. A number of visual systems for out-of-the-window viewing of the real world has been delivered to the Air Force and Navy. A digital visual system is being built for the UH-60A (Blackhawk) helicopter, while digital radar simulator systems, digital visual systems and digital electronic sensor systems are being built for the B-52 aircraft simulator. This technology is evolving rapidly and is directly applicable to ADSS.

As presently envisaged the ADSS will be a laboratory evaluation tool. It will consist of an executive computer which controls the overall simulation tasks, a sensor simulator that is easily programmable to model any sensor under evaluation, possibly a microwave simulator to model radar devices, a platform to move the sensor realistically, a readily modified CRT display, a data base that includes terrain, culture, moving targets, threats and ordnance, and a scoring (evaluation) console.

Its operation is envisioned as follows. The sensor parameters which describe a particular sensor will have been prepared off line from whatever data is available. This will include (but not limited to) resolution, field of view, spectral range, sensor range, PRF when applicable, type of scan, etc. This data will be loaded into the sensor simulator. The display will then be programmed via a panel to provide the correct sensor display field, brightness, etc. The control console will then be used to switch in the proper sensor platform (helicopter, fixed wing, RPV) etc. The control console also sets up the mission scenario. This will include such entries as gaming area, type of threats, programmed moving ground targets, type of offensive and defensive weapons, type of laser designation where appropriate, and type of performance scoring.

After the problem is set up the sensor operator is briefed as to the type of his sensors and the range of his responses. He can then be realistically moved over and through the gaming area either on a preprogrammed course or flown through by a "pilot". In the latter case he can communicate with the "pilot" and interact with the information from his display. During the flight he can be faced with the entire range of tactical situations: moving land targets, moving air targets, ordnance, decoys, smoke, weather, multiple targets. He may also deliver ordnance and be scored on performance. (A number of tactical scenes taken directly from a digital sensor system will be provided separately, and the titles of these photos are included as Appendix I of this final report.)

All of the features described in the previous outline of an ADSS simulator are possible features of current digital sensor simulators. To design a simulator tailored to your needs, it would only be necessary to model the moving platforms, generate the console layouts, design the sensor data bases, and possibly generate a more general mathematical model for the sensors.

Section 2 of this report describes the functional concepts embodied in the ADSS simulator in more detail, and Section 3 provides a detailed description of a possible embodiment of the ADSS device.

2. SIMULATOR FUNCTIONAL CONCEPTS

2.0 OVERVIEW

The Advanced Digital System Simulator (ADSS) device is intended to simulate the operation of the following two general classes of systems:

A. Helicopter-borne sensor systems for threat identification/neutralization, with block diagram shown in Figure 2.0-1.

and

B. Fixed-wing Remotely Piloted Vehicle-borne sensor systems for threat identification/neutralization, with block diagram shown in Figure 2.0-2.

The basic functional requirement of the ADSS device is the effective simulation of specific members of the above classes of systems, together with simulation of a selected battle area composed of terrain features, target characteristics, target actions (both threat and evasive), ownship characteristics, responses to flight controls and to threat and evasive actions, together with daily and seasonal weather effects.

The major simulated subsystem characteristics for FLIR/LLLTV sensors are:

a) system resolution effects, which are mainly sensor effects in Type A helicopter-borne systems.

b) spectral ranges of sensors.

c) sensor and display nonlinear intensity effects, such as overload, blooming, memory, etc.

d) for Type B RPV-borne systems, data compression/decompression effects, and down-link effects (particularly effects of jamming, etc.).

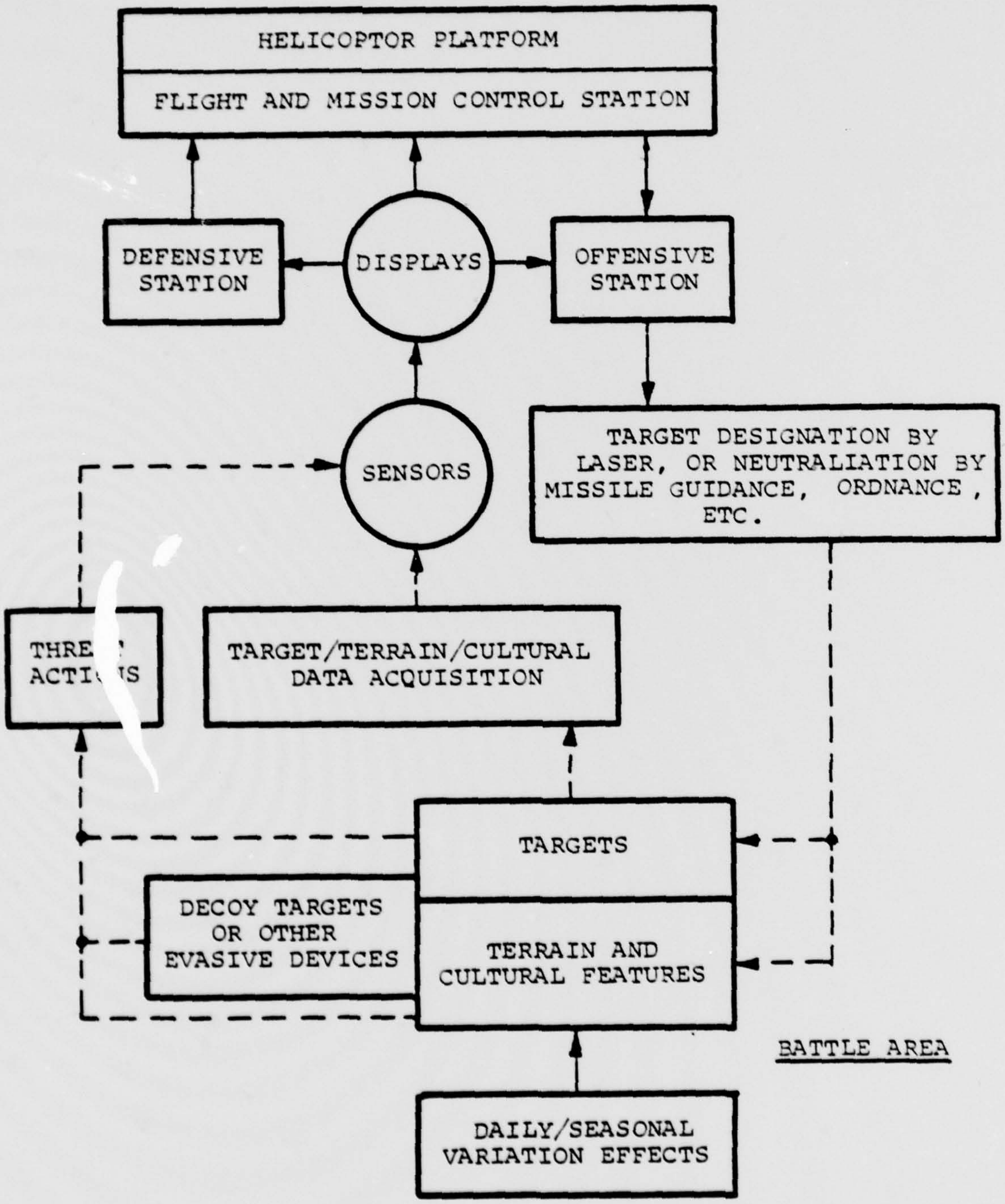


Figure 2.0-1 Helicopter-Borne Sensor Systems (Type A)

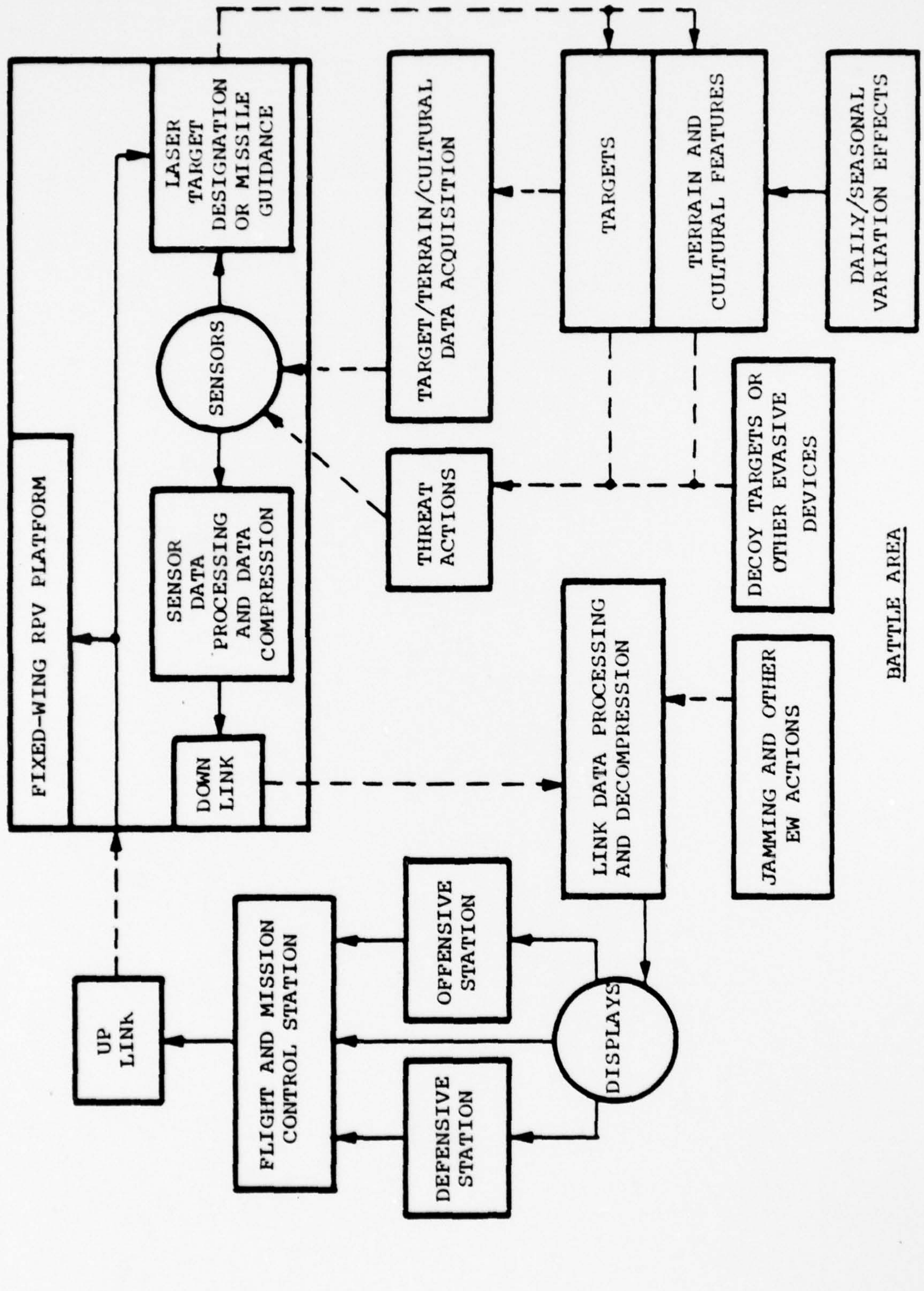


Figure 2.0-2 RPV-Borne Sensor Systems (Type B)

The major characteristics for passive/active microwave sensor systems are:

a) transmitter frequency.
b) antenna beamwidths.
c) scan rates.
d) data compression and down link effects for Type B systems.
e) direct jamming into the antenna, and other deception signals.

The utilization of the ADSS device to test proposed new members of Type A and Type B systems is proposed to be through use of Mission Scenarios, which are simulations of typical sequence of events in real-world battle areas. Each Mission Scenario needs to provide operator-selectable lists of available parameters as follows:

1) Sensor: parameters such as resolution, bandwidth, camera effects, radio link effects, (where applicable), display effects, etc.
2) Environment: parameters such as terrain and cultural data base selection, target types, numbers of targets, and actions of targets.

When these selections have been made, from a preprogrammed list of available alternates, the particular mission scenario to be used in a test sequence will have been determined. New additions to the lists of available parameters can be added as desired through a Data Base/Scenario Modification Console. Two major classes of scenarios can be implemented:

1) Preplanned: here the actions of the targets follow a routine which is independent of the actions of ownship platform. A typical case would be where the targets are designated or tracked, but the existence of ownship is not made known to targets.

2) Adaptive: here targets do become aware of ownship, and react through use of ordinance, evasion, etc.

By repeated passes through the same scenario with identical or equivalent test personnel, while modifying environment parameters, particularly target actions, with sensor system characteristics held fixed at the initially selected parameter values, it is believed that field performance of a system of given sensor characteristics can be determined. Further test passes could then be performed while modifying sensor parameters, with environment and target characteristics held fixed at the final scenario values of the preceding passes, to provide a preliminary determination of useful sensor characteristic improvements.

2.1 System Architecture

The Advanced Digital System Simulator (ADSS) can be envisioned as a complex of interconnected special purpose computational subsystems which produce simulated responses for microwave and infrared sensor displays. Simulation of low light level TV can be provided if desired. These sensor responses are displayed in real time, and vary as a function of changes in the settings of the operational controls of the simulated vehicle. A subsystem typically includes a general-purpose digital computer, provided with memory modules for storage of operating programs, mission scenarios, and data bases, and also provided with suitable peripherals, operator controls, and data communication interfaces to other subsystems. The subsystem also may include high-speed special purpose image computing devices, image display devices, and operational instruments and controls associated with the simulated vehicle.

A possible ADSS architecture is shown in block diagram form in Figure 2.1-1. This system is modular in nature, and those subsystems which are considered optional in an initial configuration are indicated in dashed blocks. A further degree of modularity cannot be shown on this simplified block but exists in reduced general-purpose computer memory and peripheral requirements for a beginning configuration. Space and interface connections will be provided for these devices and they can be added as needed as the system is expanded toward a final configuration.

The mission management subsystem contains the mission management computer complex, the operational controls for the simulated vehicle, and a suitable cockpit or enclosure for the test subject, the controls, and arrangement for convenient viewing of the CRT display monitor. These elements are shown as existing singly, as would be implemented for the minimal initial system, but could well be required to be duplicated if more realistic RPV and Helicopter control modules were implemented later.

The Infrared (IR) Image Generator Subsystem is required in the basic system, as shown. Small modifications of the Sensor Digital Data Base will permit the inclusion of data applicable to low light level TV, several IR bands, and even extremely high-frequency microwave bands. Probably the first option to be added would be Offensive/Defensive controls and display generator for simulation of laser designator or missile guidance. The microwave image generator subsystem is then implemented by a Digital Radar Landmass Simulator and a Radar Effects Generator.

It should be noted that the optional radar special effects generator shown on the figure possibly would not be required, since radar effects often can be included in the DRLMS proper. A typical DRLMS is usually optimized to simulate a particular radar, but a unit built for test purposes such as ADSS can be

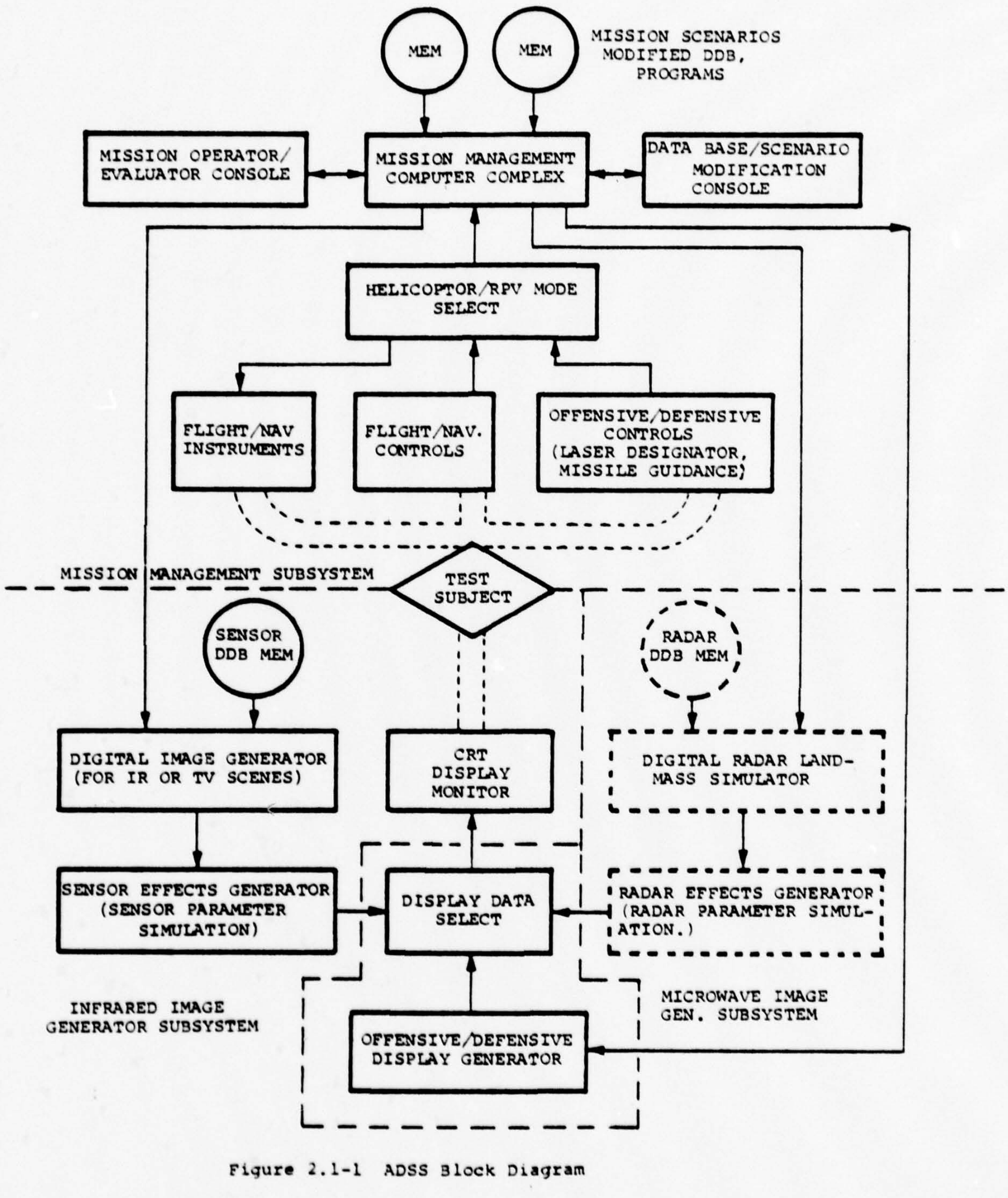


Figure 2.1-1 ADSS Block Diagram

provided with modifiable tables and interconnections for the modification of such factors as pulse length, beamwidth, atmospheric effects, etc.

The question of whether the DIG IR scene generator also could generate a radar output signal may arise. This has been tried and it has been found to be difficult to get realistic pictures for X and K band radars since the "face" structure of the DIG IR data base persists in showing through, making the radar scene unrealistic. The DRLMS data base for such radars usually uses a "grid" structure which permits the generation of realistic simulated radar pictures. This is true for X or K band surveillance/attack radars operating over relatively long ranges. For future short-range, high-resolution, target seeker radars (possibly M band) it is conjectured that the face structure of the DIG IR scene generator would be much less objectionable, but this mode of operation has not as yet been tested.

It appears reasonable that coherent IR radar devices now in the laboratory phase would be simulated in the DRLMS, rather than the DIG which is more suitable for camera-type devices.

Several other sensor configurations can also be simulated by the proposed device. For example, if color monitors (either shadow-mask or beam-penetration CRT) are provided, synthetic color can be readily added. In this case, color is typically a function of the IR or Radar return intensity.

Such displays also can provide simultaneous representations of the relative intensity of returns from two or more IR bands, such as 3-5 and 8-14 micrometer windows. Similarly, simultaneous representations of the responses of an associated radar can be displayed. Such displays may permit more certain target identification. For example, a tank might show a strong radar return

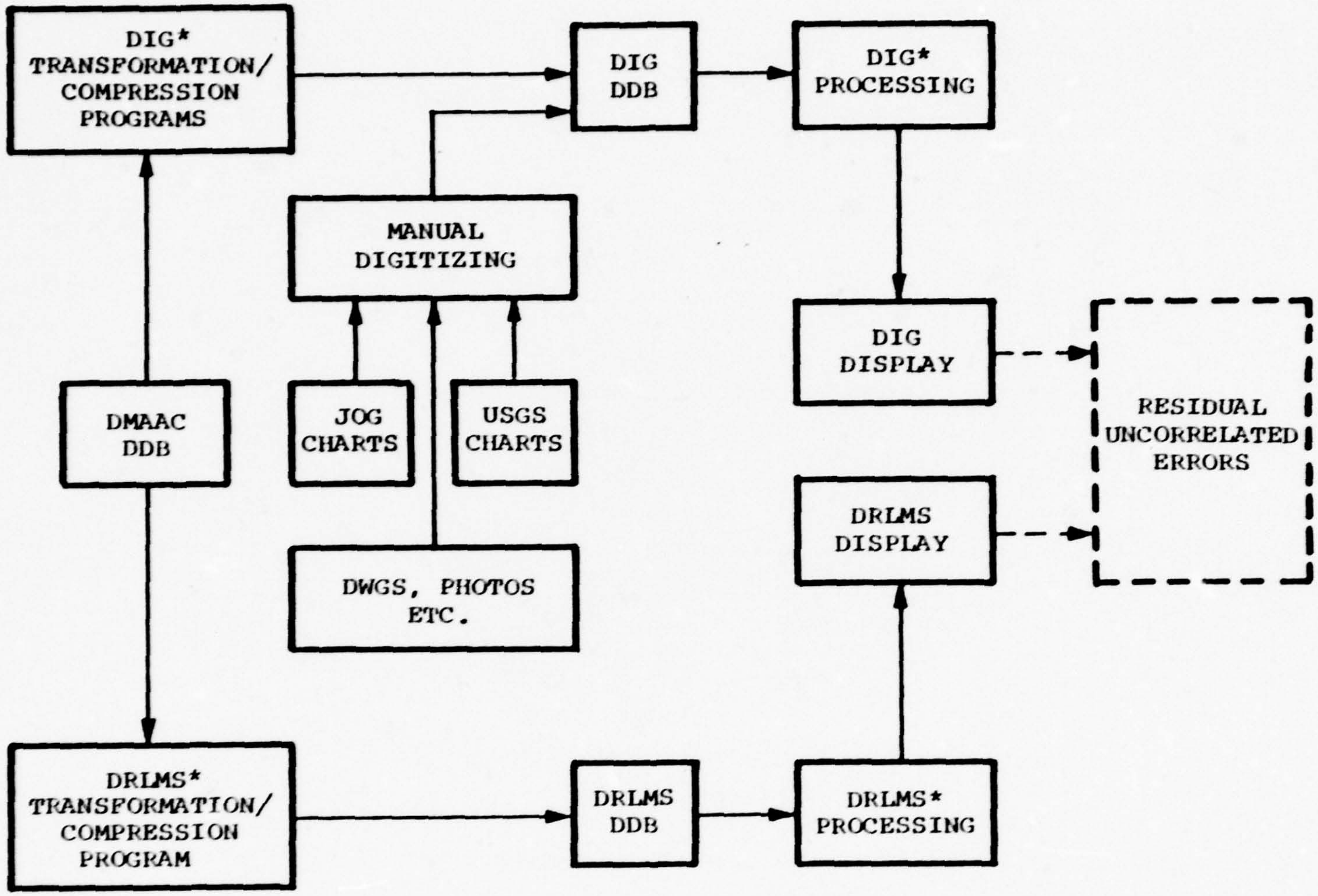
and then coupled with the proper combination of emissions from several visual and IR windows, characteristics of engine heat, gun heat, ambient light, etc. the probability of expending effort on low-threat targets, or decoys would be reduced.

2.2 Radar/Sensor Correlation

In some operational systems an observer may have the capability of viewing both an IR and a microwave display simultaneously. In these cases the ADSS system must assure that common objects in each sensor image appear at the same geographic location on each display.

It is presumed this question will not arise in the initial minimal ADSS system, since only the IR (or TV) display will be implemented. Also, the initial gaming area may be quite small, for example 10 x 10 nm. Only very high frequency quasi-IR types of microwave sensors could be used in such a limited region. If, however, an expanded version of the ADSS device was provided with the DRLMS type of microwave image generation subsystem, and the gaming area expanded to typical radar-scanned terrain regions, for example 100 x 100 nm, then correlation of the two sensors becomes important.

For such larger gaming regions, the IR DIG and DRLMS data bases can be generated from a common source, such as the DMAAC DDB, using somewhat similar techniques. If this is done there will be some degree of correlation between the two displays which are generated from the two data bases. The important question is the degree of correlation that can be expected, and the measures that can be taken to ensure that this correlation is as close as possible. Figure 2.2-1 illustrates the operations usually required to generate the two data bases. Those operations marked with an asterisk introduce differences between the data base being produced and the source material. Since these operations are not



* INDICATES A PROCESS WHICH IS A SOURCE OF UNCORRELATED ERROR

Figure 2.2-1 Data Base Generation Processes

identically implemented for the IR sensor and DRLMS data bases some uncorrelated error will result.

In a following section, after the DIG and DRLMS devices have been described, radar/sensor correlation effects will be described in more detail, and uncorrelated error minimization techniques considered.

2.3 Display Loading

The Infrared Image Generator Subsystem, using the Digital Image Generator device is subject to two types of display loading effects. These are edge density limiting and retrieval rate limiting. The cause of edge density limiting is the requirement of inclusion in the digital data base of detailed cultural complexes or heavily broken terrain. These are described in terms of faces or edges, and at some degree of complexity the capacity of the high-speed special-purpose image computing circuitry will be reached. The cause of retrieval rate limiting is the maneuvering of the simulated aircraft at low altitude at sufficient velocity that the capacity of the data base object retrieval circuitry will be reached, as new objects enter the field-of-view.

In a following section, after the hardware and software design of the DIG device for IR image generation has been described, the design precautions to control display loading will be described in more detail.

2.4 RPV Considerations

For the RPV-borne systems an additional complication of simulation in ADSS exists. As shown in Figure 2.0-2, this type of real-world system requires radio links, with typically a narrow-band high-power command up-link serving several RPV systems, and a narrow-band vehicle telemetry down-link (which may be time-

division multiplexed from several RPV systems) together with a wide-band imaging sensor down-link unique to each RPV, which transmits the output of on-board FLIR, LLLTV, or microwave sensors. Such down-links are susceptible to several types of jamming or spoofing as well as to natural transmission losses and interferences. In their efforts to minimize this susceptibility, the equipment designers use several techniques which tend to modify, and degrade the image seen at the ground control station. Thus the helicopter-borne systems of Figure 2.0-1 have only the camera and local monitor errors, while the RPV-borne systems have these plus the radio-link effects.

Figure 2.4-1 is a block diagram of a typical RPV down-link generator. The controls which permit modification of the system performance are indicated. First, the camera control unit will permit adjustment of the number of frames per second, and the number of picture elements per frame. This will degrade the picture quality but will reduce the data bandwidth requirements for the link. A control on the A/D unit will permit adjustment of the number of gray levels per picture element further reducing bandwidth. This quantized and digitally coded video data could be transmitted directly, but would be vulnerable to jamming, etc., thus further steps are taken to protect the data. The example shown is the generation of the Hadamard transform of the data, and selection of certain low sequency Hadamard coefficients for transmission. The term sequency is a descriptor of the number of sign changes in the coefficient. Alternatively, predictive coding algorithms could have been used. Other transform-domain processes are available, such as Fourier, Walsh, Karhunen-Loeve, Haar, etc. Walsh is effectively identical to Hadamard. (A large literature exists in this area and the following articles may be of interest to the reader: "Image Data Compression by Predictive Coding" H. Kobayashi and L. R. Bahl, IBM J. Res. Develop. March 1974, p 164-179. "Hadamard Transform Image Coding", W. K. Pratt, J. Kane, H. C. Andrews, Proc. IEEE Vol. 57 No. 1 January 1969,

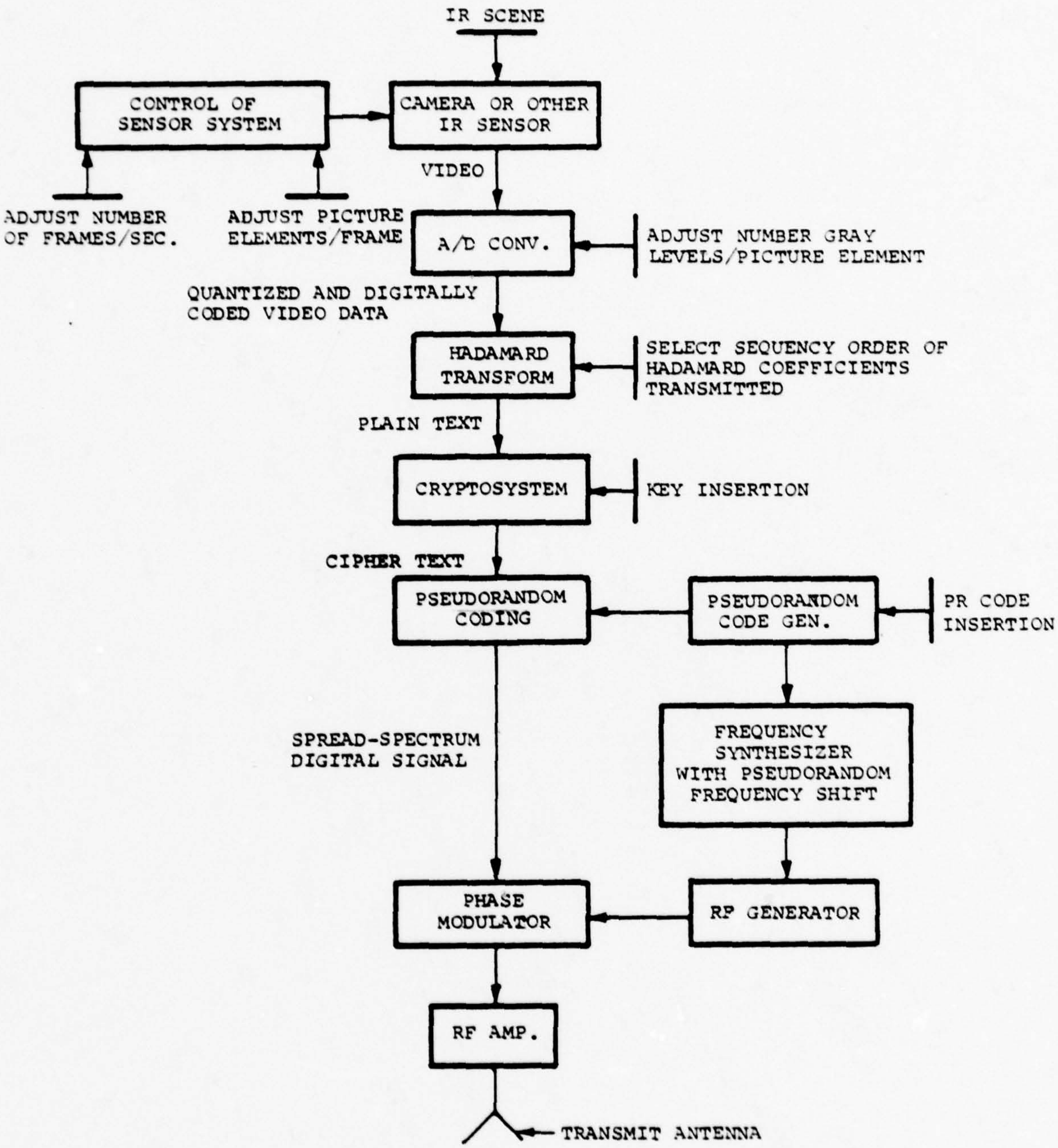


Figure 2.4-1 Typical RPV Down-Link

p. 58-68. "Real Time Video Compression Algorithm for Hadamard Transform Processing", S. C. Knauer, IEEE Trans. on Electromagnetic Compatibility, Vol. EMC-18, No. 1, February 1976, p. 28-36. "Design of a Synchronous Walsh-Function Generator", L. C. Fernandez and K. R. Rao, IEEE Tran. on E. Comp. Vol. EMC-19, No. 4, November 1977 p. 407-410)

The use of Hadamard-Walsh transforms may be dictated by the simplification of hardware allowed by the fact they require only additions, rather than multiplys and adds needed for Fourier transforms. The reduction in effectiveness of the transform compared to Fourier is described in "Sinusoids versus Walsh Functions", N. M. Blachman, Proc. IEEE, Vol. 62, No. 3, March 1974, p. 346-354. Hadamard seems to be preferred over predictive coding because a channel error degrading the transformed data affects all picture elements to a small degree, rather than possibly obliterating a few, but important picture elements, in the predictively coded process.

To reduce susceptibility to insertion of false messages into the links, (a more sophisticated form of jamming) a cryptosystem may be used. Modern cryptosystems use one-way functions as described in "New Direction in Cryptography" V. Diffie, and M. E. Hellman, IEEE Trans. on Information Theory, Vol. IT-22, No. 6, November 1976, p. 644-654.

To reduce susceptibility to "barrage" jamming processes, pseudorandom coding of the basic (possibility one megabit) data stream by the output of a pseudorandom code generator (possibility 50 megabit) is indicated on the figure. Also pseudorandom carrier frequency agility is obtained when the final spread-spectrum digital data signal is phase modulated onto the RF carrier, amplified and transmitted. Typical systems obtain processing gains (interference resistance) of the order of 30 db through this spread-spectrum procedure.

Also a number of books are available for reference. Examples are "Picture Processing and Digital Filtering", T. S. Huang, Editor, Springer-Verlag; "Picture Bandwidth Compression", T. S. Huang and O. J. Tretiak, Gordon & Breach, "Transmission of Information by Orthogonal Functions", H. F. Harmuth, Springer; and of related interest, "Orthogonal Transforms for Digital Signal Processing", N. Ahmed and K. R. Rao, Springer.

In comparison to down-link problems, the RPV up-link is relatively simple, due to only narrow-band requirements, and the relatively large ground-based transmitter power available. The various possible down-link adjustments and key and code insertions, shown in Figure 2.4-1, may be pre-programmed, or if, a sufficiently secure up-link is used, may be adaptively modified during RPV missions.

2.4.1 Simulation of RPV Systems

Since at least a DIG terrain data base will be available in ADSS, and possibly a DRLMS data base also, a software module can be included which will determine link signal strengths as a function of RPV position with respect to the ground control station, and also with respect to jammers or other sources of link interference. The typically line-of-sight frequencies used for many RPV links (for example Ku band) can be cut off by mountains and other terrain features, as the RPV flies a mission and these effects can be readily simulated.

Whatever type and degree of image degradation is known to exist for a particular setting of down-link adjustments can be readily simulated through a combination of software modification of terrain data access processes, and operation of hardware controls associated with the video drive systems and CRT monitors. Similarly, image degradations due to particular types of jamming or other EW activities can be readily simulated.

The IR image is completely computer generated and as such can be displayed at the actual resolution and frame rates of the simulated equipment (either in use or planned).

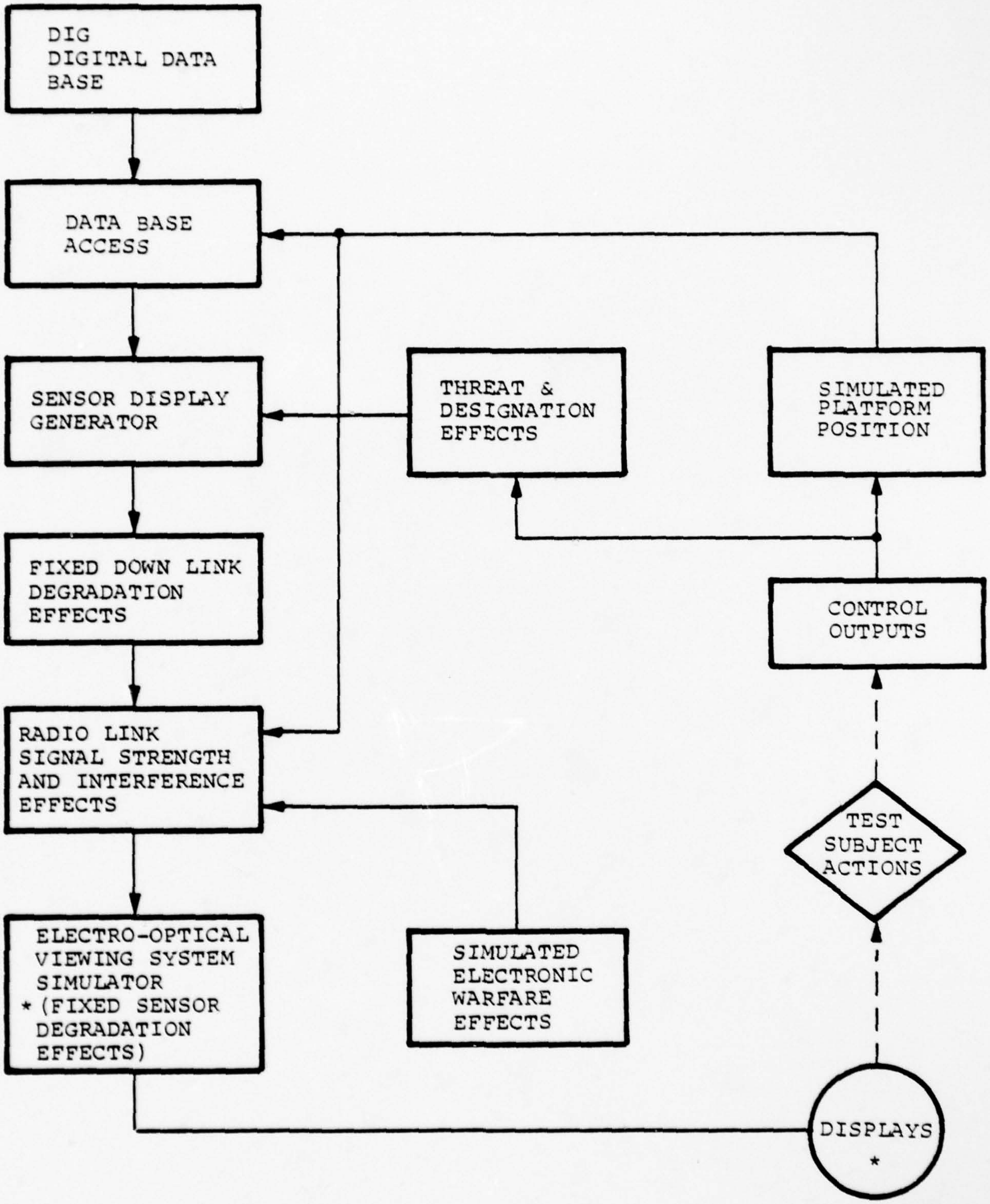
A term commonly used to describe the type of DIG device devoted to the generation of IR or TV images is the Electro-optical Viewing System Simulator (EVS). A later section will describe such a DIG device, and as a portion of this description, a subsection will provide a description of a EVS Video Effects Generator which takes the resultant IR images after whatever software-driven degradation effects are implemented, views this image through a camera, and then further degrades the image through analog modifications. It is believed that the combination of digital and analog image modification inputs can be used to produce any possible sensor, link, display special effects, including those due to EW activities. The processes used to perform these tasks are shown in Figure 2.4-2 entitled "ADSS Simulation of RPV-borne Sensor System".

The down-link generator, shown in Figure 2.4-1 composes approximately half the equipment in a complete down-link. The link also consists of simulated signal path and simulated interferences, natural and EW. Also, at the ground station, a link receiver will exist. At a minimum, the receiver must permit the inverse transformation of all effects produced in the down-link generator. For example:

- o Inverse Hadamard Transform
- o Decoding of encrypted messages
- o Synchronization of receiver, and demodulation of spread-spectrum transmitted signal.

It is also desirable to improve signal/noise ratios, to a maximum, against interference and jamming transmissions. An interesting reference indicating the state-of-the-art in this area

Figure 2.4-2 ADSS SIMULATION OF RPV-BORNE SENSOR SYSTEM



*HARDWARE MODULES. ALL OTHERS ARE IMPLEMENTED IN SOFTWARE

is "An Adaptive Array in a Spread-Spectrum Communication System", R. J. Compton, Jr., Proc. IEEE, Vol. 66, No. 3, March 1978, pp 289-298.

These receiver characteristics also can be readily simulated through additions to the same software and hardware complexes which permitted inclusion of the other image degradations. A further software module can simulate the loss of receiver synchronization due to interference peaks, etc.

It will be seen that the ADSS cannot fully predict the performance characteristics of untried equipment. The purpose of the ADSS is to simulate new equipment of known characteristcs, to permit tests by trained operators to perform specified mission tasks with this new equipment, with which they are unfamiliar. These tests, of course, also can include new jamming or path interference effects, as well as new target camouflage and evasive effects.

3. ADSS DESCRIPTION

3.0 OVERVIEW

As shown in Section 2, the ADSS device is envisioned as two subsystems in a minimal configuration, and four in a maximum configuration. This section provides a much more detailed description of suggested configurations of these subsystems. These descriptions have been excerpted from data regarding devices with known operating characteristics. It may very well be that upon examination of these characteristics, by Night Vision Laboratory personnel, a simplified version of certain of these characteristics will be deemed sufficient for the purposes of ADSS. This would be desirable in reducing costs and complexity of the system. For this report, however, it was felt best to describe subsystems of the maximum capability which can be constructed without excessive development effort, and then let the users define the simplifications of capability which can be allowed.

3.1 Mission Management Subsystem

As shown in Figure 2.1-1 the Mission Management Subsystem is composed of an Executive Computer Complex (the Mission Management Computer and associated peripherals), a Mission Operator/Evaluator Console, a Data Base/Scenario Modification Console, and instruments and controls for operation of the ADSS by the test subject.

3.1.1 Executive Computer Complex

The executive computer complex, shown in block diagram form in Figure 3.1-1 is an example of the maximum capability which could be possibly needed in ADSS. If the mission scenario parameters for sensors and environment could be sufficiently reduced, and if the Operator/Evaluator Console and Data Base/Scenario

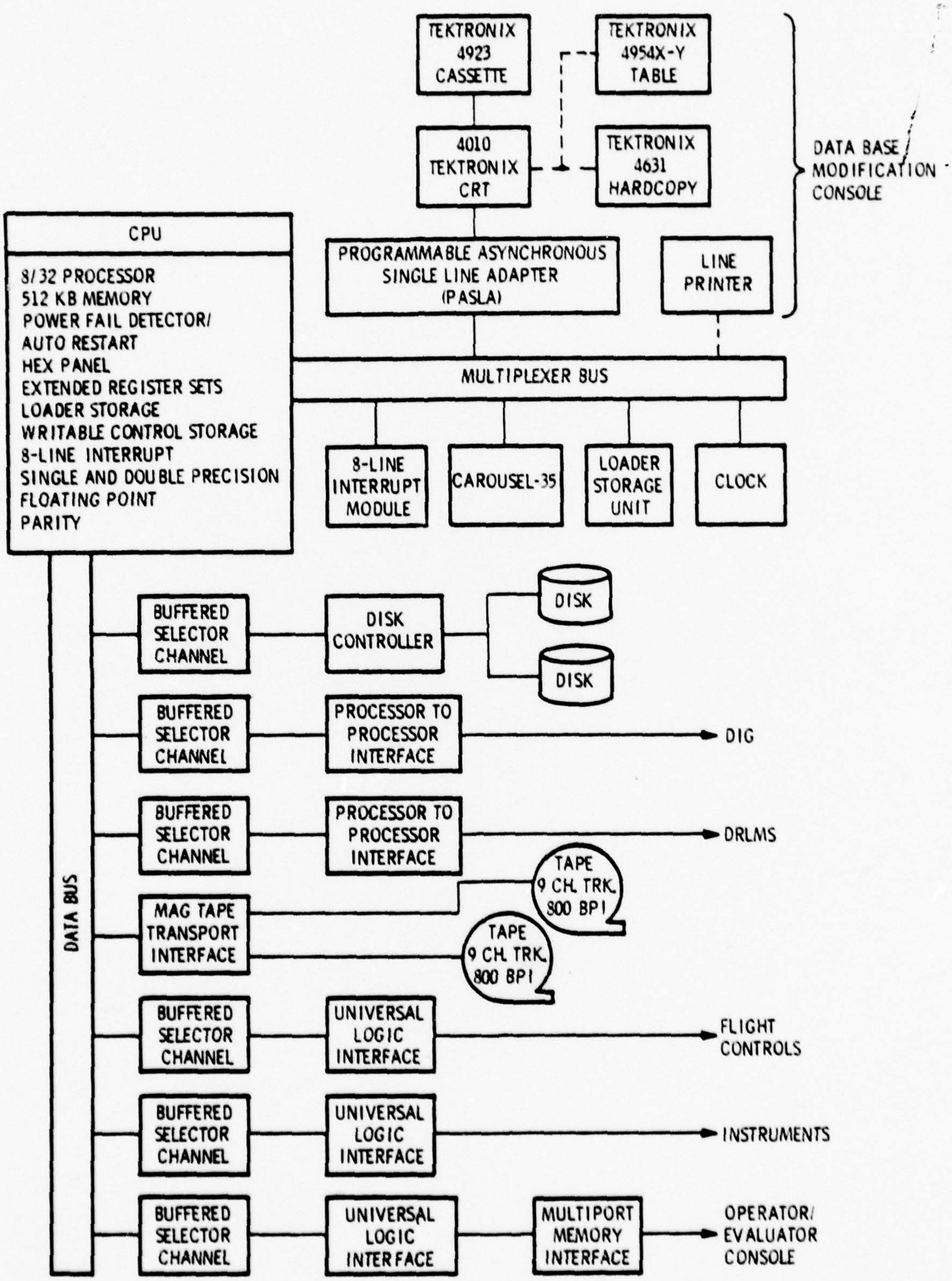


FIGURE 3.1-1 EXECUTIVE COMPUTER COMPLEX

Modification Console could be of reduced complexity, a simplified version of this computer complex could be specified, possibly not even using the indicated computer main frame, and most certainly using reduced amounts of memory, and reduced numbers of peripheral devices.

Also, as discussed in Section 2.1, if a beginning minimal configuration were implemented, using this computer type, the complex probably could be reduced as follows: Mainframe memory cut to 256 KB, one disk memory removed, one tape deck removed, three buffered selector channels removed, two universal logic interface units removed, and one processor to processor interface. If the Data Base/Scenario modification process could be simplified, at some cost in operator convenience, the 4954 X-Y table and 4631 hardcopy unit could be removed. Similar effects would exist for any mainframe and peripherals selected, as minimal to maximal configurations are considered.

The computer complex, illustrated in Figure 3.1-1, consists of an Interdata 8/32 processing unit, a total of 128K words* of core memory, and several peripheral devices. The basic peripheral units include two 80-megabyte moving-head cartridge disk systems, a Carousel 35 Keyboard Printer Terminal, and data base generation peripherals. All of these peripherals are manufactured by Interdata and are standard equipment on LINK DIG computer systems. The data base generation peripherals, whose use is discussed in Section 3.6, are also standard on LINK DIG systems. They include a Tektronix Model 4010-1 Computer Display Terminal, a Tektronix Model 4631 hard copy unit to provide copies of information contained on the Tektronix CRT, and Tektronix Model 4923 Digital Cartridge Tape Recorder. The tape recorder will enable the operator to generate commands via the 4010-1 Terminal to update the visual data during a mission, and record the operations on tape so they can be read by the computer at a later time to perform the actual updates. The Computer Display Terminal

*Another 128K words are in core memory unit.

and Hard Copy Unit will largely utilize software routines that have been developed on previous programs.

The Interdata 8/32 general-purpose processor is a high-performance, 32-bit, fully parallel device capable of directly addressing 1,048,576 bytes (one megabyte) of main memory; 512K bytes of memory are included with the CPU. The CPU features instruction look-ahead stacks and four-way interleaved 750 nanosecond memory to yield an effective 32-bit cycle time of 300 nanoseconds. The basic processor includes two sets of 16 general purpose registers (each 32 bits wide), 1024 hardware interrupt levels, 16-slot chassis, system cabinet, two power supplies, binary display panel, power fail detection/automatic restart, privileged instruction detection, and up to 1024 automatic drive channels.

The Interdata 8/32 provides for absolute as well as relocatable addressing. Immediate or absolute addressing includes 16- and 32-bit fields offering a range of ±16 kilobytes to a full megabyte. The central processing unit (CPU) includes memory access and protect controller and provides for program protection (execute, write write/interrupt, limit, and non-present protection) and program segmentation and relocation. Sixteen 32-bit segmentation registers are provided. A current loop interface to a teletype writer or teleprinter is also included. Important features not found on other machines include a writable control store, eight floating-point registers, double precision floating point hardware, a 14-port shared memory system, double indexing, and a high-speed I/O processor. Environmental operating conditions cover temperatures from 0 degrees to 50 degrees C, humidity content to 0 to 90% noncondensing, vibration conditions of 0 to 55 Hz at 0 to 125 G, and a storage temperature from -55 degrees C to + 85 degrees C.

Software packages are vital elements of the 8/32 system. Included as part of the standard software package are a real-time operating system called OS/32-MT, a FORTRAN VI compiler with optimization features and re-entrant code, MICRO and MACRO assembly languages, and an extensive support and diagnostic software complement.

3.1.1.1 Equipment Descriptions

1. Single/Double-Precision Floating-Point Unit. This unit includes a complete set of floating-point instructions plus eight 32-bit hardware floating-point registers. Typical floating-point multiply and divide times are 3.72 and 5.38 microseconds.

2. Hexadecimal Display Panel. This panel includes an advanced hexadecimal light-emitting diode (LED) readout and hexadecimal input keyboard. It includes a key-operated ON/OFF/LOCKOUT switch.

3. Universal Clock Module. This module includes a programmable precision interval clock with both frequency and interval count under hardware control. The module also includes an AC line frequency clock.

4. Interrupt Module. The eight-line interrupt module can accept level or transition inputs. Separately programmable mask and queue registers are included. The module is used to interface external interrupt lines to the built-in 255-line processor interrupt system.

5. Loader Storage Unit (LSU). This controller with hardware watchdog timer and integrated-circuit sockets, controls up to 16 128-byte bootstrap storage modules.

6. OS/32-MT LSU Bootstrap.

This 128-byte module is used with the loader storage unit. It provides an automatic bootstrap facility for loading the Operating System from disk into core memory. Additional Modules have been provided for magnetic tape devices.

7. Universal Logic Interface.

The interface includes fully buffered logic by byte and halfword data transfers on multiplexer bus or selector channel. The unit can mount up to 77 14- or 16-pin dual in-line package (DIP) integrated circuits for custom design. It incorporates wire-wrap stakes for circuit integration and includes internal cable.

8. Programmable Asynchronous Single Line Adapter. (PASLA)

This unit provides asynchronous operation for a 103/202 data set or local RS-232 terminal. The adapter operates at 75 to 9600 baud, in full or half duplex mode.

9. Removable Cartridge Disk System.

The disk transfer rate is 1.2 million characters per second, and average access time is 38 milliseconds. The system includes a disk drive controller for up to four drives, disk drive, disk pack, and power supply to support two disk drives.

10. Magnetic Tape Transport Interface.

This interface controls up to four industry-compatible continuous read-after-write 45-ips drives, and includes cyclic redundancy check hardware and read-after-write check.

11. Magnetic Tape Expansion Transport.

This is a 9-track, 800-cpi, 45-ips magnetic tape unit with a continuous transfer rate of 36,000 characters per second.

12. Buffered Selector Channel.

This channel provides the 8/32 processor with a direct memory access of up to one million bytes of main memory. Includes burst/halfword mode selection, 16 halfword buffering and burst length select from 2 to 14 halfwords. Data transfer is at rates up to 5.9 million bytes per second. Accomodates up to 16 device controllers.

13. Writable Control Store.

The WCS is a high-speed memory that can be altered by the programmer to contain routines that require frequent iterations. 512 bytes are provided; a maximum of two kilobytes is available. It has a 240-nanosecond instruction time.

14. Model 8/32 General Purpose Processor.

The central processor unit (CPU) is a high performance 32-bit fully parallel device capable of directly addressing 1,048,576 bytes of main memory; 262,144 bytes of memory are included with the CPU. The CPU features instruction lookahead stacks and interleaved memory to yield an effective 32-bit cycle time of 300 nanoseconds. The processor includes two sets of 16 general-purpose registers (each 32 bits wide), 1,024 hardware interrupt levels, 16-slot chassis, system cabinet, two power supplies, binary display panel, power fail detection/automatic restart, privilege instruction detect, and up to 1,024 automatic driver channels. The CPU includes memory access and protect controller and provides for program protection (execute, write, write/interupt, limit and non-present protection) and program segmentation and relocation. Sixteen 32-bit segmentation registers are provided. A current loop interface to a teletypewriter or teleprinter is also included.

15. Core Memory.

The Interdata 750-nanosecond core memory will be used to form the first 262,144 bytes of memory, which is the basic minimum configured with the CPU. The memory chassis and power supplies will be provided by Interdata to support the additional quantity of 262,144 bytes of core memory.

16. Processor-to-Processor Interface.

The processor-to-processor interface (PPI) is a full-duplex, high speed (500 kilobytes per second) intercomputer data transfer channel. It will be used to connect the executive and DIG computer complexes and allow complete independence of each computer complex by way of a simple disconnect.

17. Parity Generator. Includes parity upgrade and check for basic CPU memory.

18. Switching Power Supply.

Switching regulated power supply for the writable control store and single/double processors floating point unit.

19. Multiport Memory Interface.

The 15-inch circuit board provides an interface between the CPU, and ESELCCH, or Special interface and to-or-from eight multiport memory banks. It will be used to provide a rapid transfer of data directly from the disk to the active data base memory of the image generator.

20. Current Loop Interface.

Provides an interface between the CPU and the Carousel 35 Terminal.

21. Carousel 35 Keyboard Printer Terminal.

This unit prints 30 characters per second using a 64-character ASCII subset with an 132 -character print line. The terminal

operates via a 20 milliampere current loop interface, and includes external cable and friction-feed paper handling for 15-inch forms width. This unit is a buffered terminal and operates at a rate of 1320 bits per second between the terminal and computer interface.

22. Pedestal Mount.
This provides for free-standing installation of the Carousel 35 Terminal.

23. Computer Display Terminal. This terminal consists of a direct-view storage tube with a TTY-style keyboard featuring 63 printing characters and standard ASCII control characters. Three operating modes can be selected from the keyboard or the computer: Alphanumeric, Graphic Display, and Graphic Input. In alphanumeric mode 35 lines of 74 characters each constitute a full display screen. In Graphic Display mode the terminal produces clear, accurate vector displays in response to computer commands. In Graphic Input mode operator/computer interactivity is permitted.

24. Hardcopy Unit.
The unit provides permanent dry copies of any information displayed on the 4010-1 terminal screen. For greater flexibility it can also be multiplexed to make copies from up to four display terminals and/or display monitors. The 4631 uses 3-M Brand 7770 Dry-Silver paper to give the high image contrast needed for complex graphics and alphanumerics. The 4631's copy time is 18 seconds for the first copy, and 10 seconds for subsequent copies.

25. Digital Cartridge Tape Recorder.
The recorder is compatible with the 4010-1 Terminal. Each tape cartridge can hold 200,000 characters in high-density storage. Each data file has a variable number of formatted records.

The tape recorder operates at approximately 10 kilobaud.

26. Graphics Tablet.

The Graphics Tablet allows rapid digitizing from drawings placed upon the 30 X 40 inch tablet using the digitizing pen. This tablet is fully compatible with the 4010-1 Terminal system.

27. Card Reader Interface.

This interfaces the card reader to the computer system.

28. Card Reader.

The card reader operates at 1000 cards per minute and includes an external cable. The unit has a 1500-card hopper capacity and 1500-card stacker capacity.

29. Line Printer Interface.

This interfaces the line printer to the computer system.

30. Line Printer.

The fully buffered line printer operates at 600 lines per minute and provides 132 columns with a 64-character set. The unit includes an external cable.

31. Disk Drive.

A removable-cartridge disk drive will be provided which has the same characteristics as the drive in the basic system and shares the same controller.

32. HT/2 Remote Computer Control Unit.

The HT/2 is a portable hand-held interactive terminal which is functionally equivalent to a teletypewriter.

3.1.2 Computer Complexes in Other Subsystems

The DIG device in the Infrared Image Generation Subsystem, and, if used, the DRLMS device in the Microwave Image Generation Subsystem, also incorporate computer complexes. Figure 3.1.2-1 shows a possible configuration for the electro-optical visual system computer complex. A second very similar complex would be required for DRLMS. The considerations described in Section 3.1.1 regarding simplification of the computer complex if mission scenario requirements are reduced, also hold, but not to as great an extent, since this computer complex must control the production of IR images in real time. These complexes are mentioned here to emphasize the importance of selecting computers and peripherals of the same types, for ease of communication, software commonality, and improved maintainability.

3.1.3 Mission Scenarios

The structure of mission scenario files should allow the following:

1. Creation of new mission files
    a) Time required to define and layout data should be minimized.
    b) Programming to implement the data should be easy and straightforward.
2. Modification of existing missions
    a) Major modification/updates should be accomplished using capabilities of the computer's operating system and support software.
    b) Temporary on-line modification should be easily accomplished and make efficient use of time and computer resources.

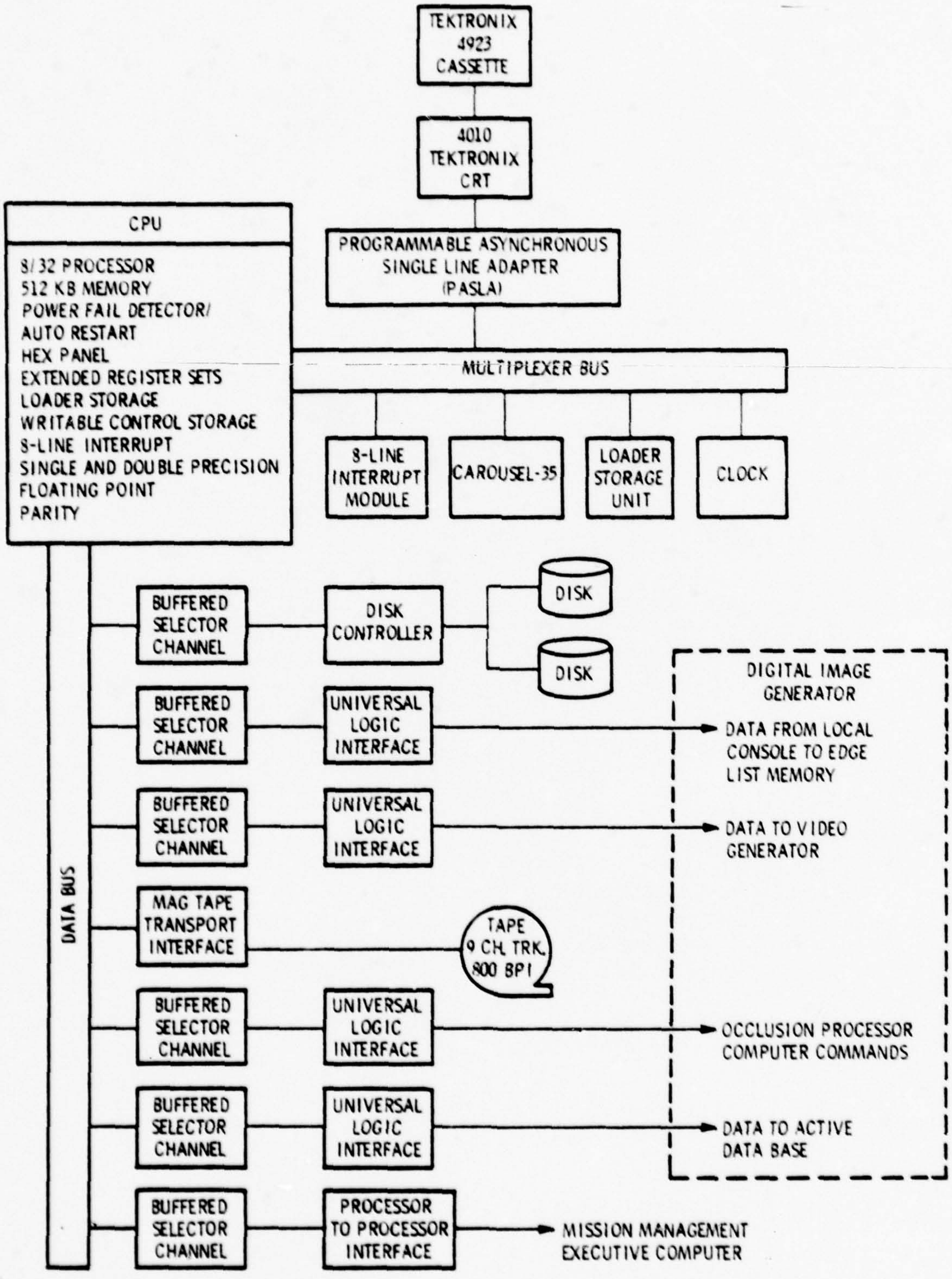


FIGURE 3.1.2-1 IMAGE GENERATION SYSTEM COMPUTER COMPLEX

3. The mission file structure and associated on-line and support software should provide capability for expansion with relative ease to accommodate future requirements.

The proposed structure of the mission files provides for these requirements and allows rapid on-line selection and ease of reviewing detailed content of a file by the instructor.

A complete mission file will be composed of one set of programs/data selected from n sets of function-oriented programs/data. For example, Mission File 1 for the Mission Management Station might comprise the following:

Mission Description	Target Type 1 Description
Initial Position	Initial Position
Aircraft State	Target Activity Description
Environmental State	Target/Aircraft Interaction Responses
Sensor System Condition	...
Mission Flight Profile	(Other Target Types, etc.
Programmed Message Set	...
Programmed Malfunction Set	
Evaluation Criteria Set	

With this functional partitioning of the mission file, efficient modular support and on-line software can be developed. For example, the programmed message trigger routines could be coded in FORTRAN as overlays, and the creation of a new set of routines would be a simple task using the FORTRAN compiler and supplied job control for updating the file. No special support software would be necessary. Also, several of the functional groups (Initial Positions, Aircraft State, etc.) require CRT pages and associated data for the Operator/Evaluator console.

The mission file pages may be created (adding a new parameter) or modified (incorporating new values) using the CRT page generator, which is a generalized program for providing alphanumeric display pages/data. On-line edit routines will be supplied to provide capabilities for temporary modification of these data sets.

A CRT page at the Operator/Evaluator console can be used to summarize essential elements of the mission scenario, such as mission objective, flight profiles of the helicopter or drone simulated, target activities, target responses, target designations achieved, malfunctions used, and relevant evaluation of ownship performance criteria.

The status page will also provide for instructor control of items that require implementation during the course of the exercise. Having control from the status page will eliminate the need for the instructor to "turn pages" during the problems and will enable him to monitor displayed parameters without interruption.

3.1.4 Operator/Evaluator Console

The console described in the following sections represent a typical configuration used successfully in training simulators. Simplification in mission scenario requirements or reduction in operator convenience could permit some reduction in complexity. However, it has been found that some sort of CRT page display seems the most effective method of operator/simulator communication and control.

Among the typical controllable items on the status page are the following functions:

- Resetting to a given program instruction number.
- Flagging segments of the simulator performance to be recorded.
- Playback of selected segment or portion thereof.

3.1.4.1 Display Format and Operating Modes

To allow maximum utilization of the CRT by the instructor to accomplish testing in the status, problem control, and communication modes, special display formats are proposed. In actuality, the instructor has one visible and one virtual (invisible) display. Controls on the panel allow quick interchange between the two presentations (Format 1 and Format 2) previously assigned by the instructor. The instructor may set up a specific presentation for reference material on the virtual display, and have the main status display (ground plots, etc.) on the visible display. The quick interchange controls allow both displays to be available without need for excessive control interaction. At any time the displays can be reassigned during a training period.

<u>CRT FORMAT NO. 1</u>

CRT Format No. 1 will be as shown in Figure 3.1.4.1-1. Each of the four sections of the format will be devoted to certain tasks. The visible CRT will normally contain a Status Display in the top four inches of the display. This data will consist of airspeed/altitude time history, malfunction status, parameter status, (freeze or out-of-tolerance), and environmental conditions status. The status display plots time history airspeed and altitude plots on the lefthand 5-1/4 inches of the CRT. The 5-1/4 inch plot represents 12 minutes of history. The vertical four inches represents airspeed and altitude (instructor selected). The center 3-inch display is divided into three areas. The topmost area is devoted to indications of device status: i.e., freeze, flight parameters frozen and/or out-of-tolerance. The center area is devoted to display of mission elapsed time and digital time (instructor-controlled) readout.

The right 3-3/4 inches of the display is devoted to malfunction and/or target indications, ten of which may be active (inserted) simultaneously. Targets or functions may be selectively removed or all active items may be removed through use of the master clear switch.

The lower 12 x 12-inch area of the CRT will contain a Ground Plot Display (GPD). Progress of the simulator within the simulated geographical area will be plotted relative to radio navigation facilities in either a game-centered or discrete-zoom aircraft-centered plot mode.

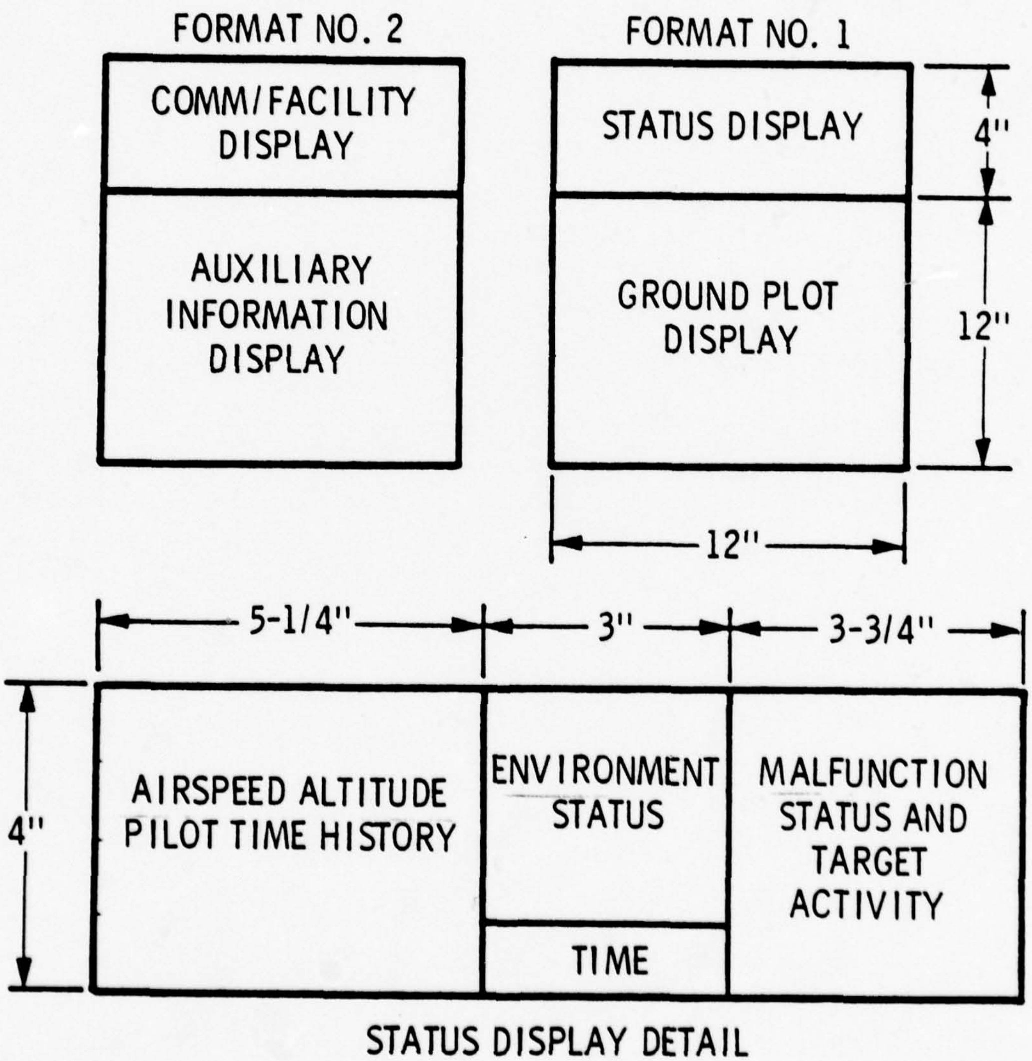


Figure 3.1.4.1-1 CRT Formats

CRT FORMAT NO. 2

The upper four inches of this display format will be a Comm/Facility Display (CF). Tabular textural data such as initial conditions, current conditions, malfunction lists, target lists and navigation station facility data will be available for instructor observation and/or modification. (See Figure 3.1.4.1-1)

The lower 12 inches of the format will be used as an Auxiliary Information Display (AID). Tabular textural data such as initial conditons, current conditions, malfunction lists, and navigation station facility data will be available for instructor observation and/or modification.

The bottom 1-inch strip will be reserved as a scratch pad for edit and error messages on pages where editing is possible.

With the display in status mode, a typical display which may be selected is a cross country plot as shown in Figure 3.1.4.1-2. Graphical plots of target activity paths, etc., also will be available on call. With the display in problem control mode tabular data will be available for instructions, observation, and/or monitor the exercise. An index will be used to present to the instructor a listing of the data vaailable.

Figure 3.1.4.1-3 presents a typical index page and provides an overview of instructor-selectable CRT pages. If the instructor already knows the number of the page that the he wants displayed, he can select that page directly without using the index. The index page has been sectioned into logical groupings to facilitate usage.

Current conditions will be depicted as shown in Figure 3.1.4.1-4 on the parameter/freeze page. This display contains current flight and environmental conditions.

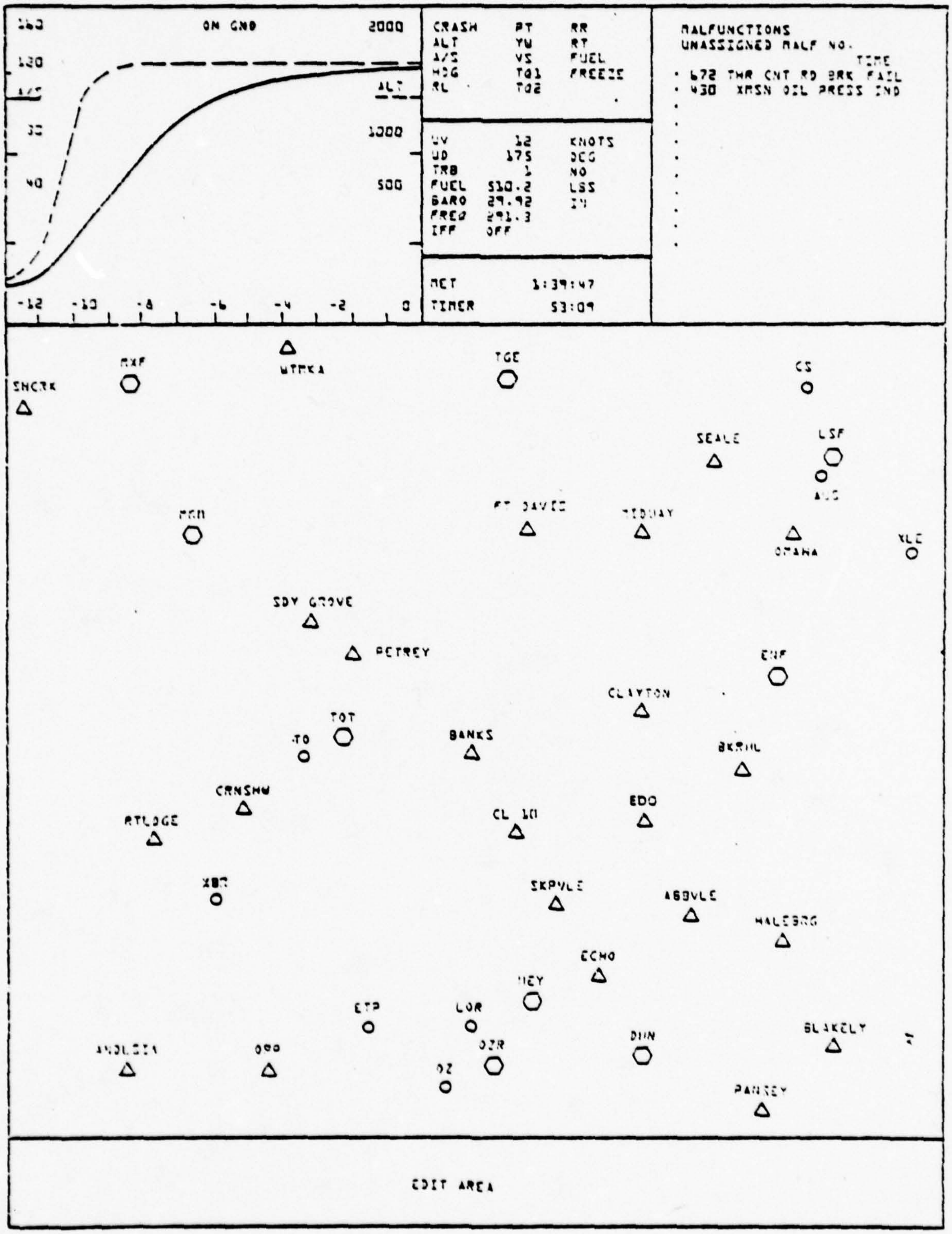


Figure 3.1.4.1-2 TYPICAL CROSS-COUNTRY MAP DISPLAY

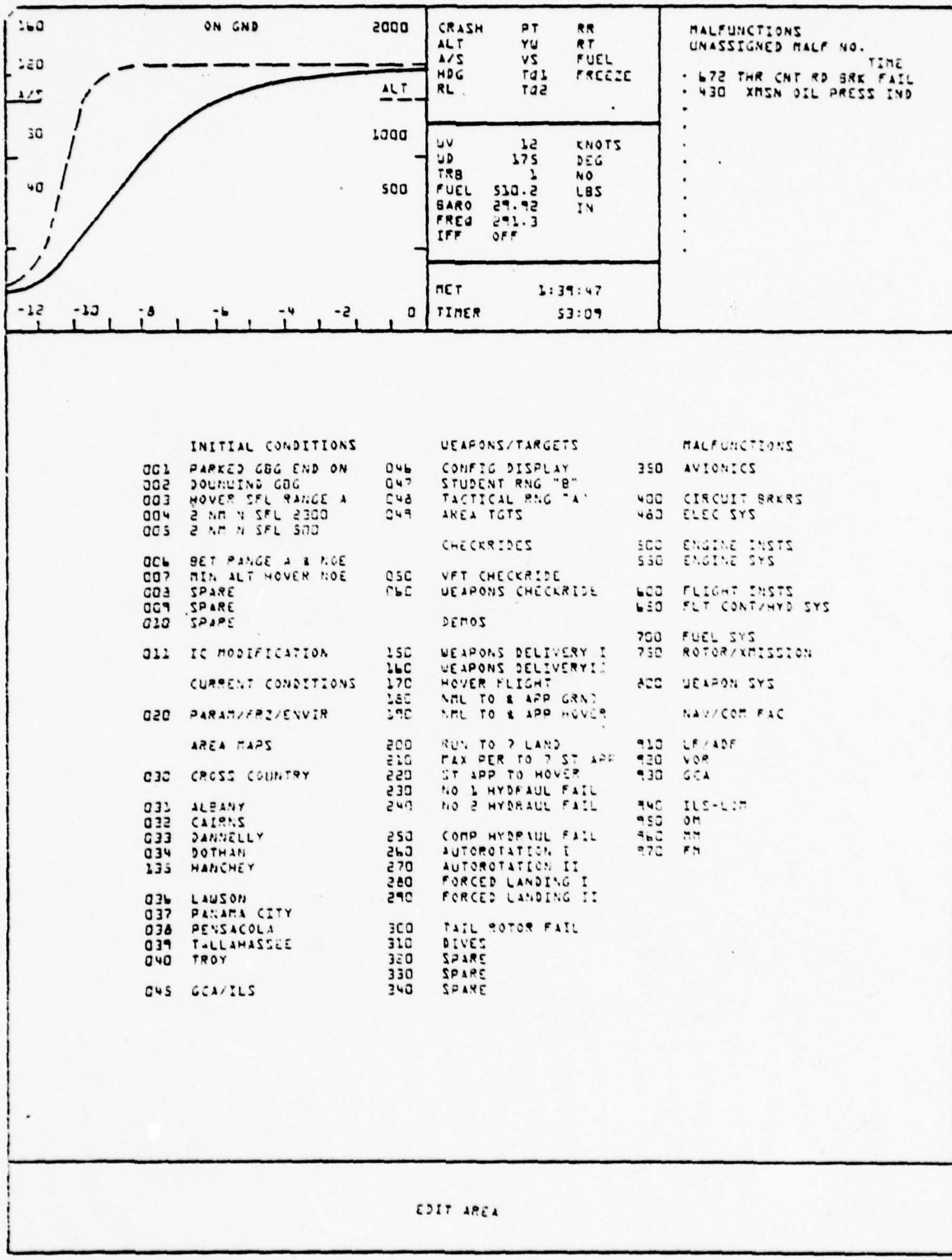


Figure 3.1.4.1-3 TYPICAL INDEX DISPLAY

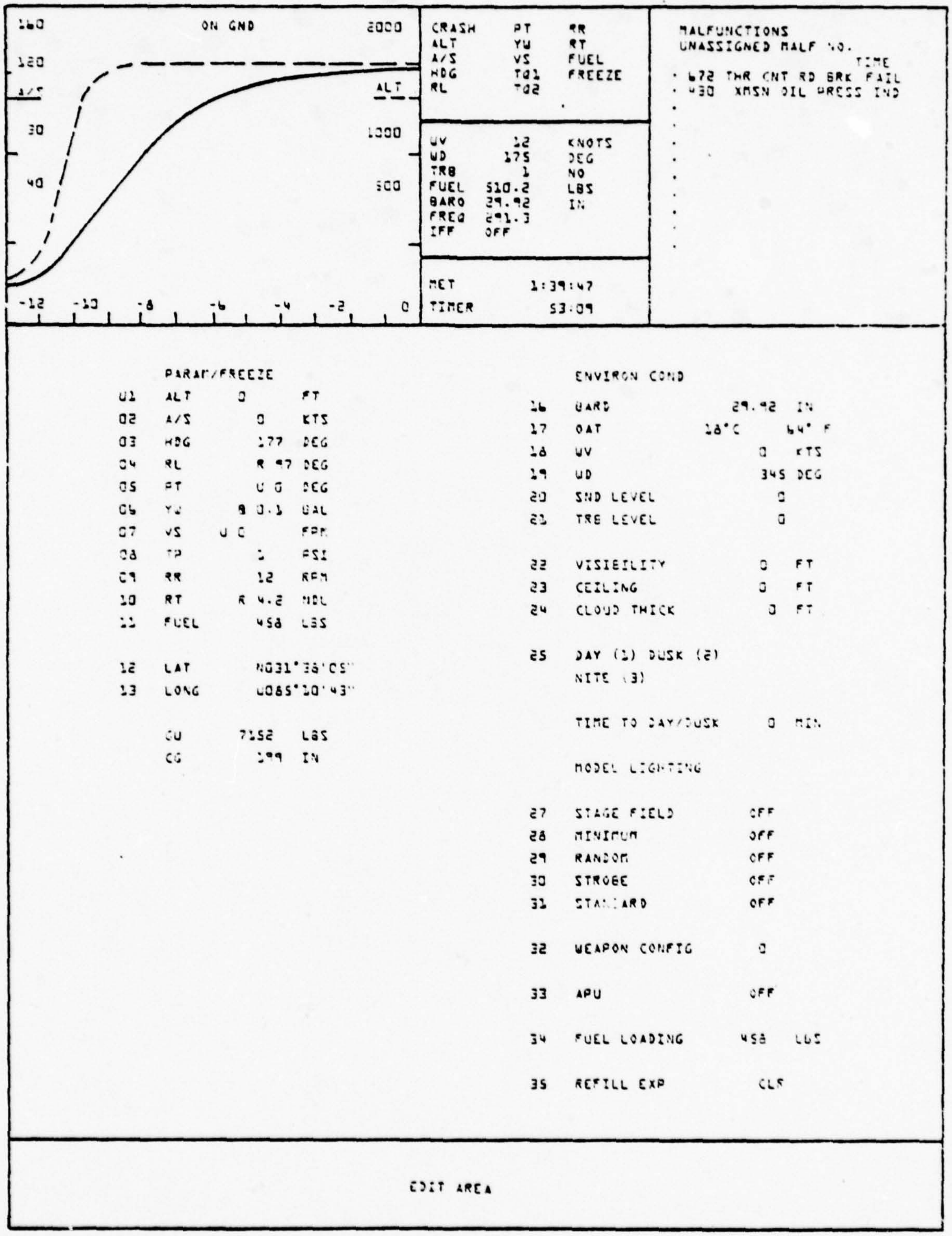


Figure 3.1.4.1.-4 TYPICAL CURRENT CONDITIONS DISPLAY

3.2 FLIGHT SYSTEMS

The drone (RPV) and helicoptor aerodynamic math models discussed in the following sections are based on the requirements of fully implemented flight simulators in which inputs to motion systems must be provided, in addition to inputs to instruments and a sensor display monitor.

For the ADSS device, which has no motion system, it may be desirable to simplify the motion equations, since the only perception of motion by the test subject will be by modifications of the flight instruments and scene changes of the sensor monitor.

3.2.1 Aerodynamic Math Model

The simulation of flight involves the determination of forces and moments which act on the aircraft due to propulsion and external environment effects, the determination of the translational and angular accelerations which result from these forces and moments, and, finally, the determination of aircraft attitude and geographical position from these accelerations. A simplified block diagram of the total flight simulation system is shown in Figure 3.2.1-1. Each of the blocks in the diagram is composed of equation sets which are described in further detail below.

3.2.1.1 Drone (RPV) Flight Performance

The frames of reference to be used for flight simulation of each drone model to be simulated are defined as follows:

1. <u>Earth Frame (E-Frame)</u>
The earth axis system is a right-handed, orthogonal triad whose X-axis is directed toward true north, Y-axis toward true east, and Z-axis toward the center of the earth. Aircraft geographic position is computed by integrating the

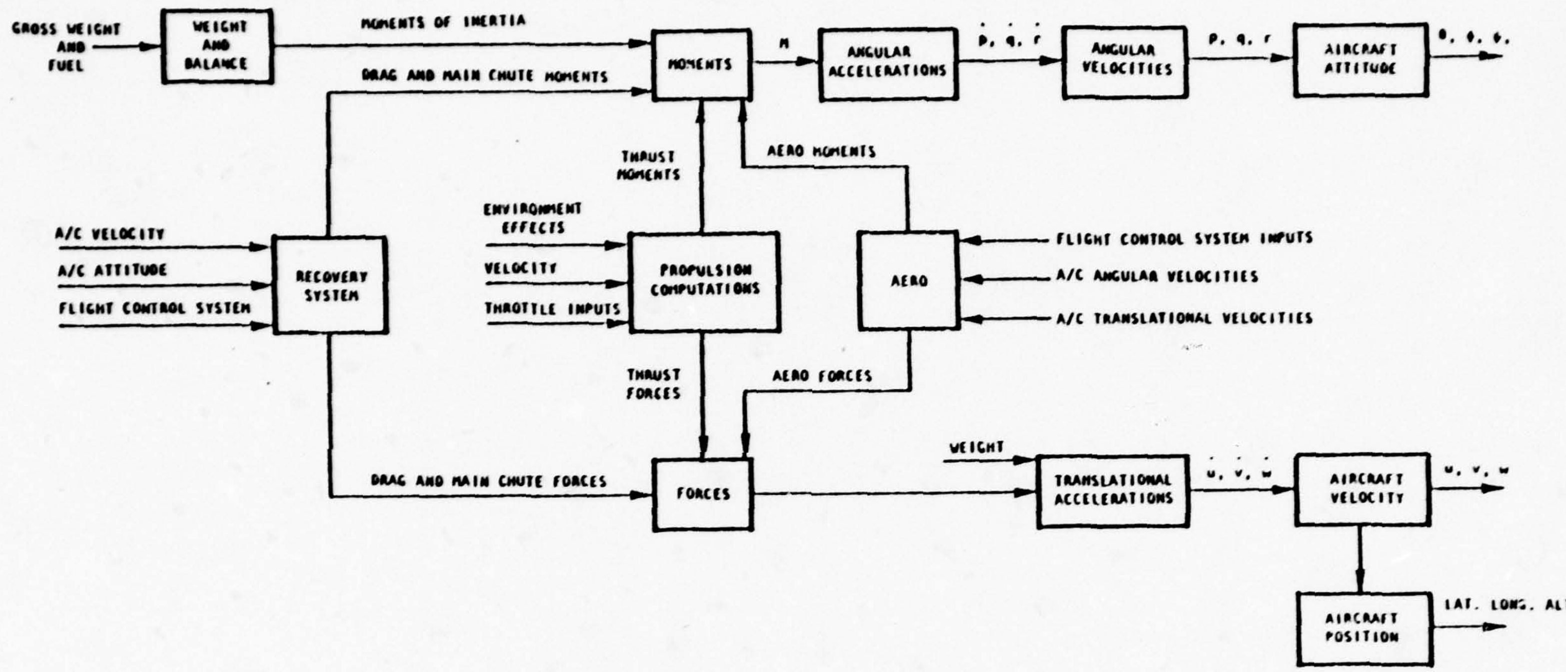


Figure 3.2.1-1 GENERAL APPROACH TO DIGITAL FLIGHT SIMULATION

three E-frame velocity components to obtain displacements which are added to the initial geographical location preset by the instructor.

2. Aircraft Body Frame (B-Frame)

The origin of the aircraft axis system is at the aircraft center of gravity. The X-axis is directed toward the nose of the vehicle and is parallel to the fuselage reference line. The Z-axis is directed toward the bottom of the vehicle, is perpendicular to the X-axis, and is parallel to the plane of symmetry of the vehicle. The Y-axis is directed such that an orthogonal, right-handed triad is formed. This axis system is the frame of reference for the rotational equations of motion.

3. Stability Frame

The origin of the stability axis system is at the point of reference for the aerodynamic coefficient data. The X-axis is along the projection of the velocity vector on the plane of symmetry. The Y-axis is perpendicular to the plane of symmetry and directed out the right wing. The Z-axis is in the plane of symmetry and is directed so that an orthogonal right-handed triad is formed. This is the frame of reference for aerodynamic coefficient data.

Aircraft movement will be described by a system of six coupled second-order differential equations, three to describe the three degrees of translational freedom and three to describe the three degrees of rotational freedom. Newton's second law of motion allows the acceleration components, expressed in some arbitrary frame of reference, to be related to corresponding components of the applied force and moment vectors. These acceleration components of the applied force and their respective integrals will completely describe the the rigid-body motion of the aircraft.

Math models describing aircraft motion can be formulated in many different ways each of which, at least theoretically, will describe the motion of an aircraft in flight. Each formulation has attendant advantages and disadvantages which must be evaluated with respect to the specific flight simulator in which they are to be utilized. The more significant areas of math model formulation related to drone simulation requirements are:

1. The simulation of the drones does not dictate a high level of fidelity for post-stall and spin. However, to assure a complete mission environment and compatibility with future expansion (addition of visual capability, more sophisticated vehicles, etc). We suggest an all-attitude simulation which has no limitations. Quaternions are utilized to define aircraft attitude; complete inertial cross-coupling is simulated; high-fidelity inertia computation is achieved by considering the empty aircraft and each fuel tank and external store separately, computing the moments and products of inertia about axes centered at the aircraft instantaneous center of gravity, again by the parallel axis theorem.

2. For an all-attitude simulation, aircraft attitude is computated most economically by employing quaternions. Computing first the quaternion rates and then integrating by quadratures avoids the singularity experienced when Euler angle rates are computed. Direction cosines and Euler angles can be computed directly from quaternions. The ASUPT (Advanced Simulation for Undergraduate Pilot Training) simulation employed an extended Euler angle formulation which was desirable because ASPUT was to be used as a research tool and it was felt that an experimenter investigating fidelity requirements would have a better understanding of Euler angles than abstract quantities such as quaternions. The ASUPT formulation required special logic and rate limiting in order to define aircraft attitude at the

singularity, and although satisfactory from a performance standpoint, was not as efficient as a quaternion formulation. Because the ADSS is a sensor testing device, the abstract quaternion formulation will be employed for its economic advantages.

3. Computing translational accelerations with respect to a fixed reference frame such as the E-frame avoids unnecessary coupling between the rotational and translational degrees of freedom. When formulating translational acceleration with respect to a rotating frame such as the B-frame many terms appear which are products of translational and rotational velocities. Although theoretically correct, this coupling tends to enhance the possibility of unstability in a digitally computed finite difference formulation of aircraft motion.

4. The angles of attack (α) and sideslip (β) are computed directly from their trigonometric relationships with the B-frame components (u_a, v_a, w_a) of aircraft velocity with respect to the moving air masses. This facilitiates direct summing of the wind gusting components, which is impossible if a wind axis formulation is employed. However, the time rates of change of angle of attack (α) and sideslip angle (β) are computed using a wind axis formulation; these are the rates which are required by aerodynamic coefficient dependencies.

The mathematical model proposed expresses the translational acceleration components in terms of aircraft body axis (E-frame) and the rotational acceleration components in terms of aircraft body axis (B-frame) coordinates. The effect of vehicle elasticity on the aerodynamic forces and moments, where significant, is considered in the conventional manner by introducing aeroelastic corrections into the aerodynamic coefficient equations.

The six differential equations, their respective integrals, and the explicit equations for the applied force and moment components comprise the flight computation system. The primary features of the model proposed are the all-attitude representation vehicle orientation afforded by quaternion representation of direction cosines, the rigor of the applied force and moment computation and the directness of computation.

The solution of the rotational degrees of freedom, provides the direction cosines required to make transformations between the E-frame and B-frame, the inputs required to evaluate the B-frame angular acceleration components and their integrals, the B-frame moment components and the moments and products of inertia. The quaternions are the integrals of the quaternion rates. The quaternion rates are functions of the B-frame rates and the quaternions themselves. Since the quaternion rates are a four-parameter system and there are only three degrees of angular freedom, only three of the quaternion rate equations are independent. The quaternion orthogonality constraint is imposed on the quaternion rate equations as an additional accuracy constraint to guard against "integrator drift".

The translational degrees of freedom are solved. The primary outputs from the translational degrees of freedom are vehicle position and velocity which are required to generate the explicit vehicle forces and moments and the instrument indications. The total E-frame acceleration components are simply the sum of the acceleration components of the various applied forces involved, because the E-frame is a proper frame of reference for the formulation of the problem under consideration.

3.2.2 Helicopter Flight Performance

The simulation techniques required for high-fidelity simulation of the helicopter's flight dynamics must address several formidable problems. First of all, there is the difficulty of simulating the rotors in real time in a fashion that provides an accurate simulation without placing unreasonable demands upon the computation system. This complexity is compounded by the interference effects of the main rotor on the tail rotor.

Second, it is essential that the means for converting flight test data - both performance and flying qualities - to coefficient form for use in the simulator math models be provided. LINK has found simulator users growing increasingly critical of simulator performance in this area in the last several years. The result is that the simulator designer can no longer allow himself to be totally dependent upon the aircraft manufacturer for design (coefficient) data for use in the simulator. He must be prepared to derive his own coefficient data based on reduction of flight test data.

In response to the first problem noted above, LINK's simulation technique to define the rotor characteristics of both the main and tail rotors will be the LINK-patented Directed Vector Approach (DVA). This method has resulted in excellent flight simulation fidelity for the UH-1H helicopter used in the Synthetic Flight Training System (SFTS), Device 2B24, and for Devices 2B31 and 2B33.

With respect to the second problem, which relates to flight test data reduction, LINK has adapted a maximum likelihood estimation program for use in helicopter simulator design*. This program (HMMLE) is run off-line and used as an aid in updating

*NASA TN D-7831 - FORTRAN program for determining aircraft stability and control derivatives from flight data.

the design to guarantee compliance with aircraft flight test data. A final update is made based upon pilot evaluation, mainly in the area of flight handling qualities.

LINK has developed a software system capable of producing an entire helicopter model from performance and test data. This system is identified as the Universal Helicopter Model System (UHMS). The major components of the UHMS and the type of information processed are illustrated in Figure 3.2.2-1.

UHMS was partially developed on Devices 2B31 and 2B33 so that the math model would be universal and general purpose in a nature, enabling it to be utilized for ADSS as well as any other future helicopter simulator.

3.2.2.1 Flight Dynamics

As an aid in organizing the presentation of the aerodynamic simulation model, the simplified block diagram of Figure 3.2.2.1-1 has been prepared. The diagram shows a total of seven sources of forces and moments which act upon the vehicle : the two rotors plus five other dynamic force and moment generators. The contributions of these various generators are summed in the two blocks labeled "Moment and Force Summations", and the total moments and forces are utilized in the Equations of Motion to compute aircraft accelerations, velocities, attitudes, and positions.

The models provide a unified, total simulation of all of the helicopter flight regimes: hover, transition, cruise, autorotation, takeoff, landing, flares, aerobatics and slingload operation. Thus simulation of each of these regimes is implicit in the context of the overall simulation math models. Special emphasis has been placed on the design for the following maneuvers:

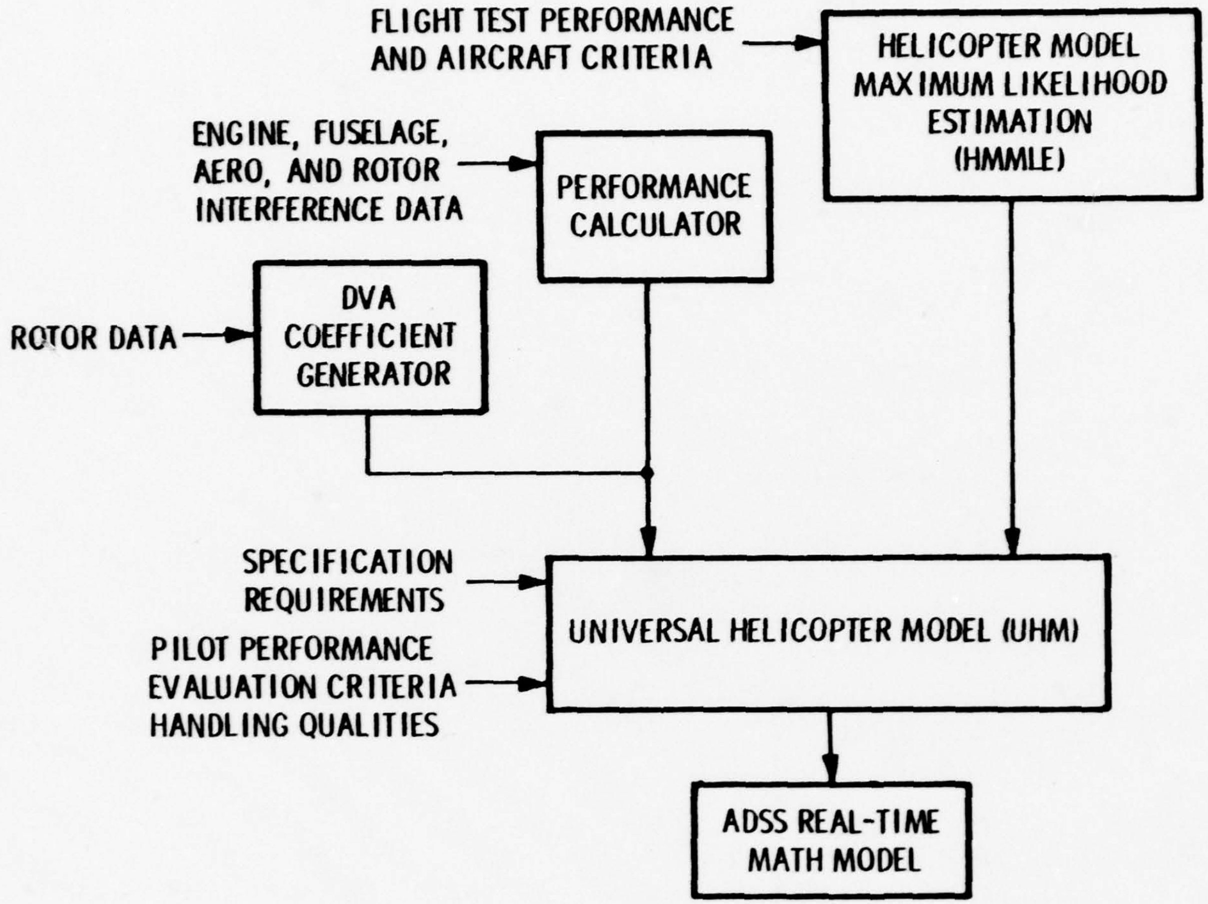


Figure 3.2.2.-1 Universal Helicopter Model System

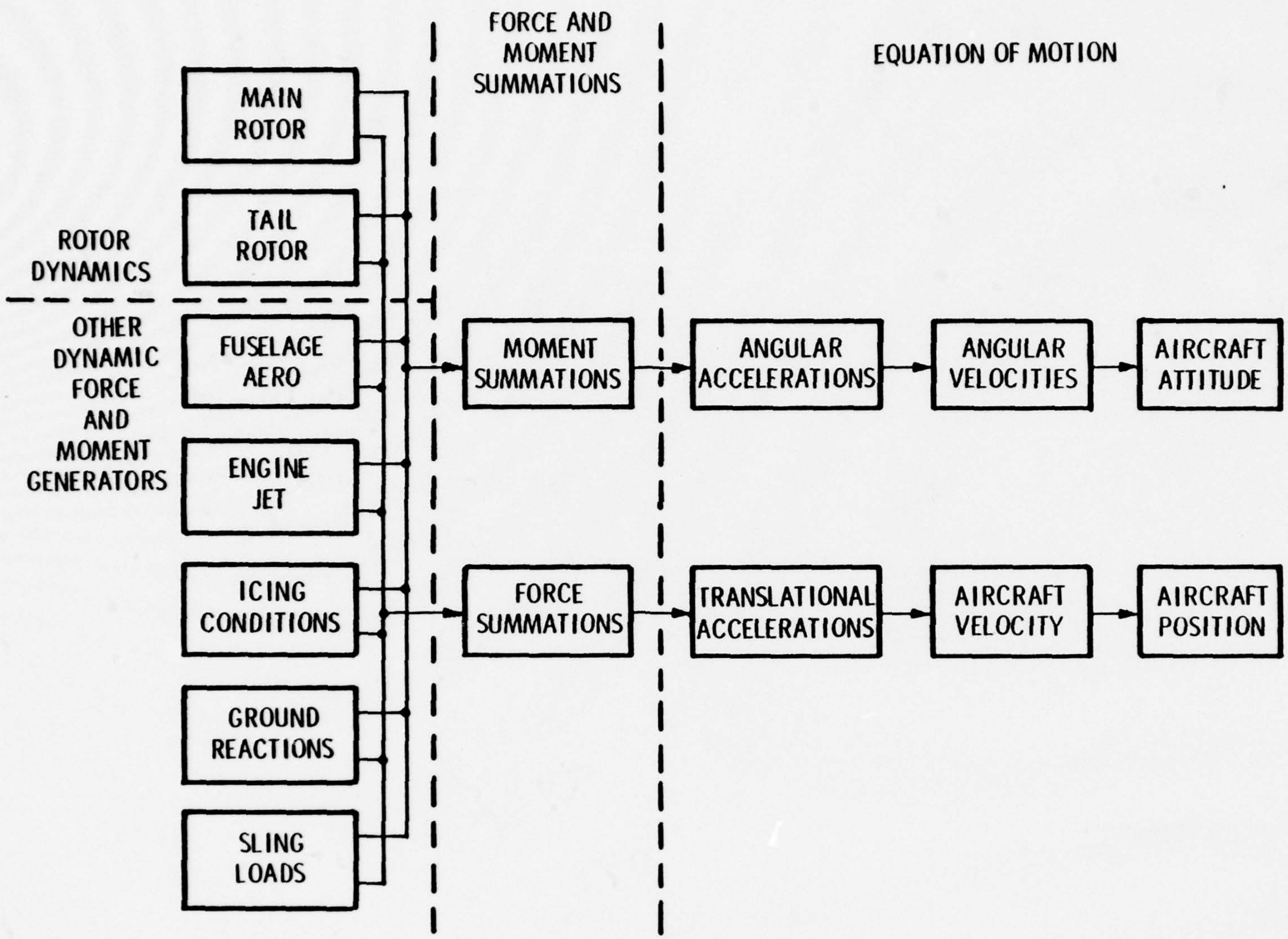


Figure 3.2.2.1-1 Helicopter Flight Dynamics

1. <u>Hover</u>

During hover the rotor is essentially stationary with respect to the air mass. Under these circumstances, much of the air which flows through the rotor is recirculated, resulting in a bound vortices system. From momentum theory, this is an inefficient situation and the rotor computation must reflect this properly. DVA also admits to this unique situation: i.e., crossflow ratio (μ) and inflow ratio (λ) are both uniquely zero and the rotor aerodynamic coefficients reflect this. No special computation is necessary during hover.

2. <u>Transition</u>

As the rotor begins to move with respect to the air mass during transition, less and less air is recirculated and the strength of the bound vortices is weakened, resulting in an increase in rotor efficiency. At forward airspeeds between approximately 18 and 22 knots the rotor sheds the vortices abruptly, resulting in a rapid increase in efficiency which is referred to as transitional lift. The abrupt shedding of the vortices also results in a phenomenon known as transitional buffet. Transitional buffet is simulated utilizing the motion system. Transitional lift occurs at unique values of crossflow ratio (μ) and inflow ratio (λ), and the rotor aerodynamic coefficients will reflect the efficiency increase appropriately with no special computation required.

3. <u>Cruise</u>

Cruise is the normal flight regime. In order to ensure that the simulator meets flight test performance, the rotor coefficient simulation must be most accurate in this area. The process by which this accuracy is obtained is discussed in the following subsections. Maximum cruise speeds require special rotor tip machnumber corrections to the rotor

coefficients, and this is included as part of the DVA.

4. Autorotation

Autorotation is a unique rotor situation in which the rotor torque requirements are near zero. This is reflected by the torque coefficient (C_Q) also being near zero. The DVA independent parameters (λ, μ, $\theta_{.75}$) are not unique and vary considerably during an autorotation from altitude to landing. This implies that the simulation of C_Q is extremely critical when the required C_Q) is in the neighborhood of zero. The relationship between thrust coefficient (C_T) and torque coefficient (C_Q) must be very precise and matched to flight test autorotation performance.

3.2.2.2 Equations of Motion

The aerodynamic performance of any helicopter is well defined by what are currently accepted as the standard equations of motion, namely the equations given in MIT Report No. 7591-R-1, "Simulation of Aircraft", dated February 1958, by Mark E. Connelly. These equations, restated here for convenience (see Fig. 3.2.2.2-1), generate the angular and translational rates as seen by the aircraft body axes and accurately describe the various phases of flight, including takeoff, landing, climb, descent, hover, level flight, autorotation, and aircraft maneuvering. The report also presents detailed discussions of the various axis systems, that are required to accurately define the relative motion and position of the simulated helicopter.

The standard helicopter equations of motion define the aerodynamic performance of the helicopter with reference to the fixed earth axes in terms of Euler angles. The model expresses aircraft motion relative to the ground and includes altitude, rate of climb, attitude, and aircraft position and ground speed. The treatment allows for computations with nonstandard day

MIT PAGE NO.	EQUATION NUMBER	REFERENCE EQUATION
38	237	$F_{xb} = m(\dot{u} - rv + qw)$
38	238	$F_{yb} = m(\dot{v} + rV_0 + ru - pw)$
38	239	$F_{zb} = m(\dot{w} - qV_0 - qu + pv)$
19	88	$A_{xb} = m(g \sin\theta_b + \dot{v}_{xb} - w_{zb}v_{yb} + w_{yb}v_{zb}) - T \cos\alpha_T$
19	89	$A_{yb} = m(-g \cos\theta_b \sin\phi_b + \dot{v}_{yb} + w_{zb}v_{xb} - w_{xb}v_{zb})$
19	90	$A_{zb} = m(-g \cos\theta_b \cos\phi_b + \dot{v}_{zb} - w_{yb}v_{xb} + w_{xb}v_{yb}) + T \sin\alpha_T$
37	234	$T_{xb} = I_{xx} \dot{p} + (I_{zz} - I_{yy}) rq - I_{xz} (pq + \dot{r})$
37	235	$T_{yb} = I_{yy} \dot{q} + (I_{xx} - I_{zz}) pr - I_{xz} (r^2 - p^2)$
37	236	$T_{zb} = I_{zz} \dot{r} + (I_{yy} - I_{xx}) pq - I_{xx} (\dot{p} - rq)$
40	260	$p = \dot{\phi} - \dot{\psi} \sin\theta$
40	261	$q = \dot{\theta} \cos\phi + \dot{\psi} \sin\phi \cos\theta$
40	262	$r = \dot{\psi} \cos\phi \cos\theta - \dot{\theta} \sin\phi$

Figure 3.2.2.2-1 STANDARD EQUATIONS OF MOTION

atmospheric conditions and also allows for changes in aircraft stability due to center of gravity displacements caused by fuel depletion and cargo loading. Provisions for freezing the simulator motion as commanded by instructor input are made within the equations of motion.

3.3 Infrared Sensor Systems

The spectral range of infrared radiation runs from wavelengths of about 0.75 micrometers to 1,000 micrometers, with regions of particular interest from about 1.0 micrometers to 30 micrometers. For infrared sensor systems operated from ground-based or airborne platforms within the atmosphere, a number of atmospheric windows exist with infrared transmission efficiencies of up to 90% in these regions.

Infrared detectors are of two major types: thermal detectors and quantum detectors.

Thermal detectors operate by absorption of infrared radiation within a bulk detector material. This may cause a temperature change in gas pressure as in a Golay cell, or change electrical polarization of a crystal as in pyroelectric detectors. Thermal detectors are of relatively low sensitivity but are independent of the wavelength of the incident radiation and often may be operated at room temperature.

Quantum detectors operate through interaction of incident photons with electrons in a solid, operating as a semiconductor. Quantum detectors are critical of the incident radiation wavelength, and usually must be operated at reduced temperatures.

These detectors are used in two major types of infrared sensors. These are devices which use mechanical scanning of the field-of-view, and devices which use electronic scanning. A typical

example of a sensor using mechanical scanning is the Bendix Thermal Mapper. This device provides a resolution of 2.5 milliradians, a field-of-view scan angle of 120°, temperature sensitivity of 0.5°C. The detector is an indium antimonide unit, cooled with liquid nitrogen, which covers a wavelength band of 0.7 to 5.5 micrometers.

Other similar devices typically use mercury cadmium telluride detectors, cooled with liquid helium, covering a wavelength band of 8 to 14 micrometers. An example of a sensor which uses electronic scanning is pyrolitic vidicon detector, which is a vidicon camera tube with a pyrolelectric target material. This tube is scanned in a manner similar to TV, except that the input infrared radiation must be chopped, and the output waveforms reshaped, since the pryoelectric effect responds only to changes in the radiation levels. Work is in progress on solid state *charge-coupled* devices to provide camera-type sensors with electronic scanning, which do not need the vidicon tube, but use arrays of quantum detectors, together with associated state-interrogating circuitry.

3.3.1 Spectral Ranges of Interest

As described in Section 3.3, the bands within atmospheric windows which are of interest to ground-vehicle and low-level airborne IR sensors are approximately 0.3 to 1.5 micrometers (visible to near IR), 3.0 to 5.5 micrometers and 8.0 to 14.0 micrometers.

3.3.2 Sensor Resolutions

A definition of resolution for a camera type of IR sensor is the number of resolvable picture elements in the field-of-view. A typical pyrolitic vidicon camera as described in Section 3.3 has a standard 525 line TV format, thus the number of picture

elements is 525 x 700 or 367,500. Special cameras, such as the RCA QTV-8 return-beam vidicon have a field-of-view resolution of 5,000 TV lines, in a square format, for a total of 25×10^6 picture elements. At present this tube provides response in the visible and near-IR region only.

A definition of resolution for an electromechanical scanner type of IR sensor is the scan angle resolution in milliradians, typically one to 2.5 milliradians for present systems. This will normally provide 500 to 1,000 scan lines if a TV type display is used.

3.3.3 IR Simulation Algorithm

Such an algorithm will be a specialized solution of the basic IR radiation equation combined with an atmospheric transmittance equation:

$$N = \epsilon\sigma T^4/\pi \text{ w cm}^{-2}\text{sr}^{-1}$$

Where N = IR radiance of the object

ϵ = emissivity

σ = Stefan - Boltzman constant $5.67 \times 10^{-12} \text{w cm}^{-2}(^\circ K)^{-4}$

T = Temperature (degrees Kelvin ($^\circ$K)

$\text{w cm}^{-2}\text{sr}^{-1}$ = Watts per square centimeter per steradian

It appears that solution to this equation is appropriate for IR simulation. However, the determination of the emissivity and temperature terms for the wide assortment of objects to be expected in an IR data base is a complex task.

The two variables in this equation are emissivity, ϵ, and temperature, T. Emissivity is the ratio of energy emitted by the specified object to the energy which would be emitted by a black body of the same temperature, T. The emissivity of an object may be determined as a function of the material of which it is composed, or of which each part is composed. The basic

value must be modified as a function of the surface finish and the shape of the object. For exposed cultural objects, weather effects will greatly modify emissivity. For example, one or more surfaces of the basic object may be coated with rain, ice, snow, soot, or other atmospheric precipitants. Similarly, temperature, T, is a function of sun (as the major energy input source) angle modified by internal heating of the object and weather effects. (Weather effects being defined as: cloud cover, falling rain or snow, current ambient temperature at the object.) The temperature is further modified by recent history, since most objects of interest have long thermal time constants, and store results of preceding weather, sun illumination, internal heating, etc.

The atmospheric transmittance equation is:

$$I = I_A + (I_0 - I_A)\, e^{-R'K_1}$$

where:

- I is the observed intensity of the object
- I_A is the observe intensity of the atmosphere
- I_0 is the observed intensity of the object at zero range
- R' is the ratio of the distance to the object to the visibility distance
- K_1 scaling coefficient
- I_0 is related to object IR radiance, N, by a constant K_2.

Parameter flexibility is inherent since the visibility range, and scaling coefficients, K_1 and K_2, can be assigned by program control.

Treating I as a single numerical magnitude is correct for wavelength-independent sensors, such as pyrolitic and other thermal sensors. However for photon sensors, each object should be provided with emissivity, ϵ, versus wavelength, λ, function tables

or curves, and I for each emitting wavelength then should be multiplied by a factor which describes the sensitivity of that particular photon detector at that wavelength. The summation of these results will provide the final detector value, I. Simulation of these effects is relatively simple, since each data base object can have known values assigned for each possible sensor type, and atmospheric windows to be used in the simulation.

3.4 Digital Image Generation (DIG)

Digital image generating equipment is being employed for most of the simulated visual attachments to aircraft simulators presently being designed and built by LINK. It is also readily applicable to infrared image generation. In DIG systems, all of the features in the scene are digitally modeled in three dimensions and then stored on a mass storage medium for access during real-time processing. The processing function in the DIG transforms the three-dimensional digital models into a two-dimensional representation with the correct perspective for display by projectors or CRT's.

The digital nature of DIG permits a great deal of flexibility in the modeling of the data base and in its subsequent display. DIG allows the observer to move anywhere throughout the gaming area and view along any line of sight with any field-of-view while at the same time keeping the entire image in proper perspective and focus. Both wide and narrow fields-of-view are easily attainable. Fixed and moving objects, both properly occulted, can be generated, and a wide variety of special effects are also attainable. The digital data base can be easily modified to include the effects of infrared emissions, and microwave reflectances, rather than the usual visual range responses.

However the DIG does introduce some new problems the most important of which is the limited edge processing capacity of the

system. Since the image is generated digitally by means of mathematic transformations, the number of computations which can be performed and the number of edges which can be generated, is limited by the amount of time available for processing. The present LINK DIG is capable of generating 10,125 potentially visible edges every thirtieth of a second.

To properly analyze the performance of DIG for ADSS, some knowledge of its hardware organization and capabilities is required. Figure 3.4-1 is a block diagram of a typical DIG system. The function of the General Purpose Computer is to retrieve the three-dimensional object descriptions from the online data base and send these descriptions along with vehicle position and attitude data to the Frame Calculator for processing.

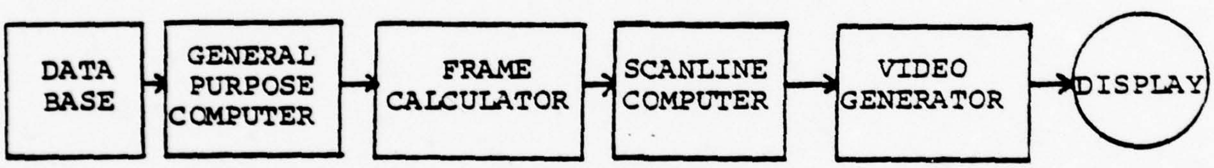


Figure 3.4-1 Typical DIG System Block Diagram

The Frame Calculator then performs a number of geometric calculations which project the three-dimensional data base onto a two-dimensional plane, clip this projected image against the boundaries of the viewing window, then organizes the resulting image into a list of edges ordered by raster lines.

This list of edges is then transferred to the Scanline Computer which determines which objects occult other objects, sorts edges from left to right along scanlines, and outputs one line of data at a time to the video generator.

Finally, the Video Generator takes data from the scanline Computer, performs smooth shading computations on designated objects to make them appear rounded in shape, applies visibility effects, removes any objectional quantization effects, and converts the digital data into video signals suitable for use in the display device.

Each of the hardware subsystems does have limited capability in performing the functions which are assigned to it. The first of these system limitations is the processing capacity of the Frame Calculator.

Within the Frame Calculator, a number of mathematical computations must be performed for each object that is processed. Vertices must be transformed from data base coordinates to window frame coordinates and then clipped and projected onto the image plane. Back-facing faces, that is, those which are facing away from the observer, must be eliminated and those faces which are retained for processing must be properly illuminated.

Even with the high-speed logic and pipeline architecture employed, the sheer magnitude of the computations to be performed and the limited frame time available, restrict the number of edges which the system can process. The LINK system is limited to 10,125 edges, but some of this capacity time is utilized for overhead operations so that the net yield is a list of approximately 8000 edges which are sent to the Scanline Computer.

The 8000 edge approximation becomes the measure of the complexity of the scene image which can be generated.

A second performance limiting characteristic, associated with the Scanline Computer, concerns the number of edge crossings which can appear along one scanline. This number is limited by the fact that the edge crossings must be sorted from left to right to facilitate the conversion from digital to video signals. The time available for this sort is the time it takes to sweep one scanline. At present this time permits a maximum of 512 intersections for each and every line.

The number of edge crossings per scanline is a statistical function of the number of edges. A plot of LINK-generated sampling points and the extrapolation of these data on a cartesian grid, where the horizontal variable is the number of scene edges and the vertical variable is the number of edge crossings, shows a high-density linear band of sampling points with scattering about this band on either side contained by an envelope. (See Figure 3.4-2). A vertical cut through this plot at 8000 edges yields a probability function which peaks at approximately 300 edge crossings and "tails off" at approximately 375 edges - comfortably below the 5% envelope (400 edge crossings per scanline).

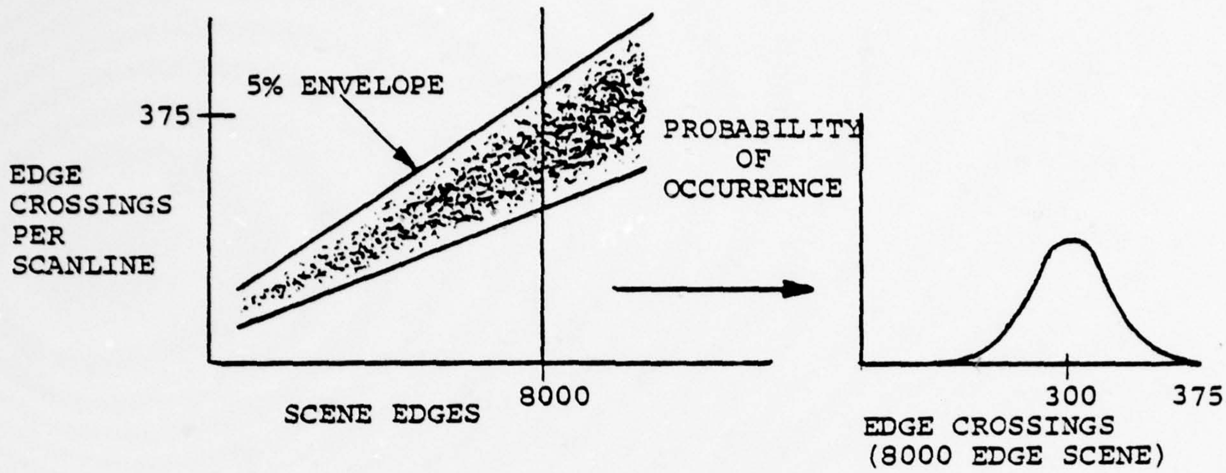


Figure 3.4-2 STATISTICAL DISTRIBUTION OF EDGE CROSSINGS VERSUS SCENE EDGES

Lastly, the Video Generator is limited in its capability to output picture elements. Current designs permit picture elements to be output at any rate between 24-45 nanoseconds per element: This permits generation of a display containing approximately 1000 scanlines, with approximately 1000 picture elements per scanline, at a rate of 30 frames/second, at maximum resolution, or the generation of any less detailed display.

The LINK DIG device, described above, is only one of many possible implementations of a Digital Image Generator. If it is assumed that a digital data base is used, then the design of any such device should include 1) Data Base Configuration Selection, and 2) Data Base Storage architecture, including real-time access procedures.

If the selected data base configuration is one using polygonal faces to approximately describe a gaming area in terms of convex polyhedra, (as LINK does) then the following algorithms will be implemented, 1) position-dependent geometric relations, 2) attitude-dependent geometric relations, 3) special illumination effects and atmospheric transfer effects. If the display device is a raster-scan CRT, then the following processes are required, 4) edge sorting, 5) occulting, and 6) video scanline generation.

However, if the selected data base configuration did not use polygonal faces, then a quite different set of algorithms might be used. For example, if the data base was in the form of a digital approximation to a hologram of the gaming area, then the digital calculation of the effects of an incident coherent reference beam originating from the simulated sensor viewpoint, and an inverse FFT, would produce information needed to generate an image of the gaming area from current platform position and sensor viewpoint.

If, however, the data base is assumed to be of the polygonal face type, then a subsidiary DIG implementation problem is to determine how to generate images of increased complexity. The standard DIG image complexity limit was described as 8000 visible edges and 5% (400) edge crossing per scanline. This limit was a result of computer processing time requirements. An obvious implementation would be to make a faster computer, since, to a first approximation, the image complexity is a linear function of computer speed.

The computer speed increase can be done in two ways, 1) use faster components in present architecture, or 2) modify architecture to improve parallelism of computations.

To modify architecture, the obvious approach is to provide several pipeline units composed of frame calculator, scanline computer, and video generator rather than one. The allowable scene complexity limit will increase approximately linearly, and so will cost, to a limit of about four times present complexity. At about this point, the general purpose computer and the real-time data base memory access procedure will become inadequate and will need redesign.

Since a gaming area described in terms of polygonal surfaces does not very much resemble the real world, an interesting data base modification would be the introduction of a more sophisticated array of objects. For example, three-dimensional objects with curved surfaces, such as cylinders and spheres, etc.; also, objects with ill-defined textured surfaces, such as trees, rock formations, etc. The required calculations would then resemble those for the standard DIG, except modified to handle the more complexly described objects. Points, lines, and polygons would still be used in the data base where applicable. This would automatically increase scene complexity limits, since many of the data base edges are presently used to delineate subpolygons sufficient to permit the use of smooth-shading processes to create the illusion of a curved surface.

3.4.1 Image Resolutions

In Section 3.3.2, sensor resolution was estimated for both IR cameras and electromechanical scanners. Typical resolutions of 500-1000 scanlines with 500-1000 picture elements per line were found. The DIG Video Generator and CRT display can provide these numbers of picture elements, however, the simulated scene still must be limited in complexity to a maximum of 8000 edges and 5% edge crossings. The actual sensors have no complexity limits except those imposed by the finite number of picture elements available, and so the only limit on real-world scene complexity is the repetitive nature of the gaming area terrain.

3.5 Digital Image Generator

The following section provides a description of a Digital Image Generator (DIG) specially oriented toward generation of Electro-optical Viewing System (EVS) images, which simulate the use of either Forward Looking Infrared (FLIR) or low-light-level TV (LLLTV) sensors.

Figure 3.5-1 is a diagram of the proposed visual simulation system. As described previously, the actual computer and peripherals used will be a function of gaming area size and system requirements. The proposed system will have the capacity to process and display more than 8000 potentially visible scene edges and lights in a single frame time (1/30 second). This total number of edges and lights may be shared in any combination.

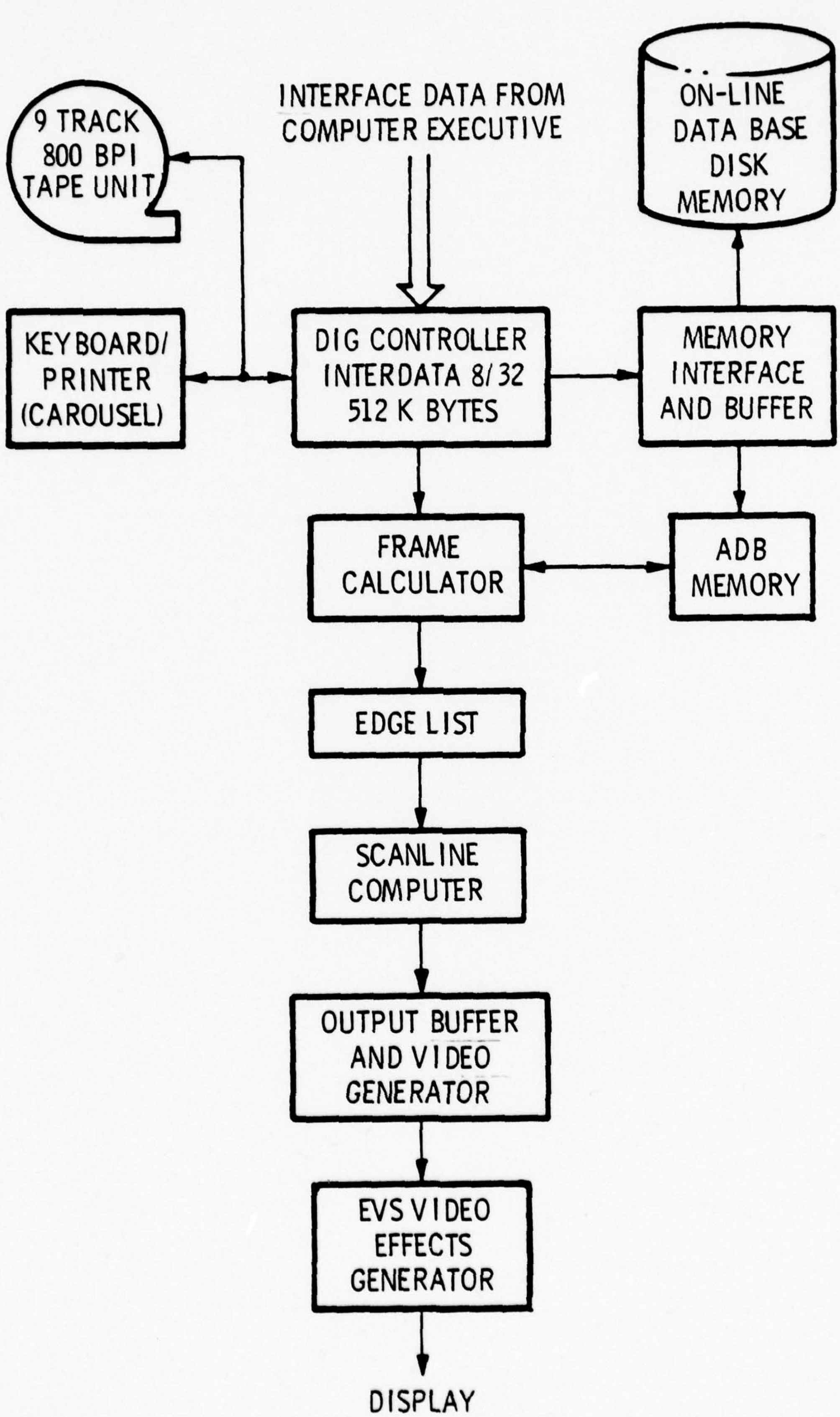


Figure 3.5-1 DIGITAL IMAGE GENERATION SYSTEM

The system can employ a two tier memory hierarchy to permit real-time access to a data base covering up to 40,000 nm^2. The top tier of storage is a moving-head disk memory.

The second tier of memory is composed of a rapid-access bulk-storage solid-state memory which holds the data in a radius about the aircraft. This memory is called active data base (ADB) memory.

It can be seen that if only a small gaming area is required, the entire data base can be stored in the ADB memory, and disk memory might be needed only for data base preparation processes.

The production of a data base of the size and complexity required for wide-area simulation depends upon the development of automatic data base generation software. The software methods are based on several years of LINK involvement in programs deriving real-time on-line data bases from DMAAC data.

In addition to the wide-area data base to be derived from DMAAC data, the system will also accept data from a graphic input data generation facility. This facility permits ease of entry (of special features not found in the DMAAC data) and modification and update of existing data bases.

The image processing hardware will consist of special-purpose processing equipment that interfaces with the disk memory, memory interface and buffer, and the cockpit displays and with the main-frame computer through direct memory access (DMA) channels.

The Image Processor consists of the ADB memory, Frame Calculator, Edge List, Scanline Computer, and Video Generator.

The data base is transferred from the disk memory to the high-speed ADB memory of the Frame Calculator through the memory interface and buffer under control of the General Purpose Computer. The contents of the ADB memory are updated as necessitated by changes in position and attitude of the input sensor. Distant objects are brought in as they come into view, and replace other objects as they go out of view. The dynamic storage allocation of objects in the memory is under control of the data base management program in the Mainframe Computer.

The geometric processing is under software control from the mainframe computer through a DMA channel. For every frame, position and attitude information for the viewpoints and all moving objects is transferred, followed by a list of objects to be processed. Occulting information accompanies the list. Occulting is dynamically determined in the Mainframe Computer from frame to frame. Illumination and visibility data are also transferred through this channel as a sun vector, diffuse illumination constant, and visibility range.

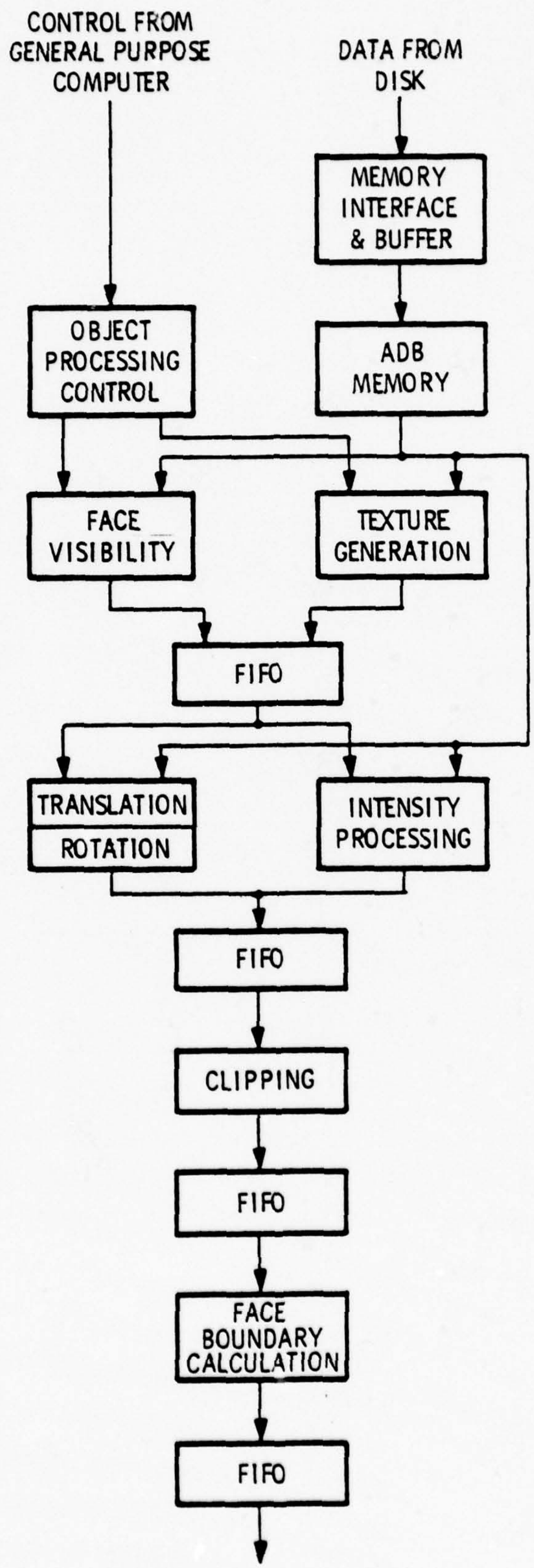
A second DMA channel is used to transfer data on the visibility of "artificial objects" which are returned to the mainframe computer by the DIG hardware.

The processing hardware is arranged in two pipelines - the Frame Calculator and Scanline Computer. There is a double buffer between the two pipelines, Edge List A and Edge List B. The result of the geometric processing is written into one, while the data pertaining to the previous frame is read out from the other into the scanline computer pipeline.

3.5.1 Frame Calculator

The Frame Calculator hardware is arranged in a pipeline, as shown in Figure 3.5.1-1. It performs all the perspective transformations on the objects that are necessary to produce a projected image. The segments of the pipeline are dedicated to subsequent steps of processing. There are first in/first out (FIFO) buffers between parts of the hardware which equalize the data flow rate when instantaneous processing rates of adjacent segments differ.

The first step in the processing is to test polygons for visibility and resolvability. If the polygon is visible, pointers to the vertices and the associated intensities are transferred to the next section of the pipeline.



3.5.1-1 FRAME CALCULATOR

3.5.1.1 Translation

Translation is the first step in the geometric transformation; the calculation of the vector V_t going from the view point to the vertex is shown in Figure 3.5.2-1.

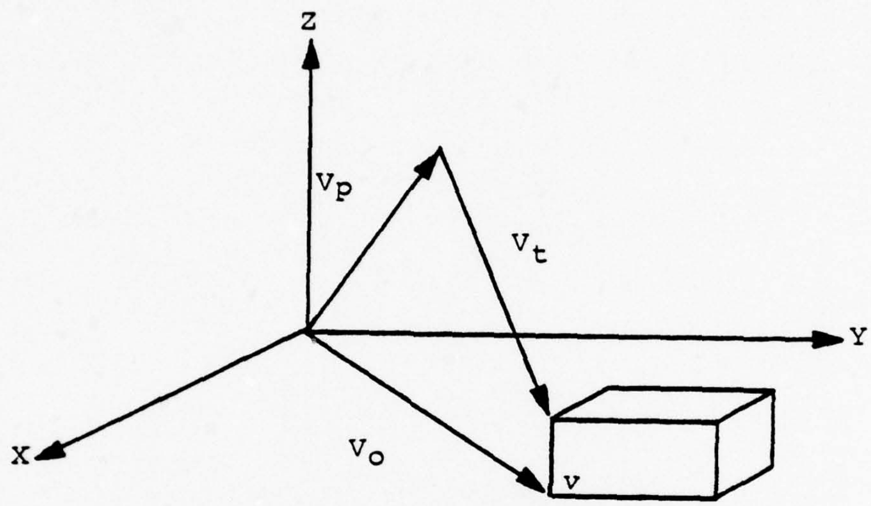


Figure 3.5.1-2 Viewpoint Translation

3.5.1.2 Rotation

The rotation hardware rotates the translated vertex coordinates from the data base coordinate system to channel coordinates. The basic operation is the multiplication of a vector by a rotation matrix H_c, which reflects the angular relationship between the reference coordinates system and the channel coordinates. Channel coordinate axis are defined in such a way that the Z_c axis is perpendicular to the projection plane. The X_c and Y_c coordinates axis are scaled so that the window has a 90 degree field-of-view in channel coordinates. The scale factors are the tangents of the horizontal and vertical half-angles of the field-of-view.

The hardware is also used to multiply matrices together, in which case the multiplicand matrix is treated as a three-column vector (H_c), which is the multiplication of the aircraft rotation matrix (H_a) and one of the five window matrices (H_w). This operation is used to calculate the transformation matrix for moving objects (H_c), which is the multiplication of the moving object rotation matrix (H_m) and the channel matrix (H_c).

3.5.1.3 Intensity Processor

Vertex intensities are calculated to show the effect of illumination or self-emission. The intensity of a vertex due to a light source at infinity is computed as a function of the cosine of the angle between the unit surface normal at the vertex (N) and the direction vector of the illumination source (S). For shaded objects the intensity calculations are performed on a vertex basis. For nonshaded objects intensities are calculated for each face. The intensity of the illumination source is variable by adjusting the magnitude of S. Diffuse illumination can also be added. The intensity calculations for simulating different times of day will utilize the capability of the software to control the direction and magnitude of the sun vector and the diffuse illumination level.

An additional function of the intensity processor is to modify the intensity of surface detail or texture faces as a function of their display size, as computed in the face visibility calculation. These faces are processed in terms of the intensity

difference between the face and the background upon which it lies. The use of intensity difference in these calculations has two benefits for the displayed image. First, it permits the contrast between the detail face and the background to be reduced gradually from the value encoded in the ADB to zero as the face becomes small at the display. This permits small detail to be added and deleted from the display without blinking in or out. It is also a very powerful method to prevent the appearance of objectionable quantization effects. The second benefit of the method is that it permits detail which lies on a shaded surface to reflect the shading of the underlying surface without increasing the complexity of the shading computation.

3.5.1.4 Clipping

The observer's field-of-view is limited by the boundaries of the window. The four planes passing through the eyepoint and the four window edges, together with the image plans, form a truncated pyramid-of-vision. Only objects or parts of objects that are within the volume of the pyramid will appear in the projected image. In the clipping process the vertices are tested to see whether they are within the pyramid.

If line segments intersect the pyramid of vision, vertices outside the pyramid-of-vision are eliminated. The intersections with the visible part of the boundary plane are substituted for vertices. When a polygon surrounds one or more corners of the pyramid, one or more vertices are introduced into the sequence of visible

vertices so that the output of "windowing" is always a closed polygon with all vertices in view.

Clipping is also used to test whether area blocks and clusters or objects are in the field-of-view. Artificial objects surrounding the area blocks can be processed through windowing and a code sent back to the General Purpose Computer to indicate if the last object or polygon is in view, out of view, or partially in view. The computer uses this information to limit the object list for each channel, thus optimizing the useful processing of the Frame Calculator.

3.5.1.5 Face Boundary Calculator

The next step in the transformation is projection. The projected vertex coordinates are obtained by division and simple scaling. Adjacent vertices of the polygons, the ith and (i-1)th, define straight-line segments of face boundaries.

The Frame Calculator has the capacity to process more than 8000 potentially visible scene edges in the 1/30th of a second available for processing a frame. As the limit of processing capacity is approached, a signal is sent to the Mainframe Computer to reduce system loading. This will be accomplished by removing information from the display in a manner which will not cause objects to appear or disappear abruptly. In the event that a momentary overload occurs, the last previously completed frame is repeated on the display. This backup method fully protects against scene breakup that might result from display of a

partially completed frame.

3.5.1.6 Edge List Memory

The Edge List Memory is a double buffer memory which permits data for one frame to be written in one section of the memory as it is output from the Frame Calculator. At the same time, data for the previous frame is read from the other section of the memory for processing by the Scanline Computer. When the Frame Calculator and Scanline Computer have completed their respective processes, the two sections of the Edge List Memory are switched. The Scanline Computer then reads the data just previously written while the Frame Calculator writes new data in the other section.

All the edges processed for a frame are stored in the Edge List Memory. In order to organize raster processing efficiently, the edges are sorted according to the scanlines (SL's) on which they start. The sorting is accomplished by linking together all the edges that start on a given raster line and storing the last link in a scanline table. There are as many lists as raster lines. When there is no edge starting on the line, the list is marked empty in the table; otherwise, the table entry points to the first edge of the list.

Edges belonging to the same object are further processed. The ones that start on the same line are sorted according to slope. Also, face boundaries that coincide are collapsed into one internal edge, thus creating potentially visible scene edges. (See Fig.3.5.1-6) The contents of the Edge List are updated every frame.

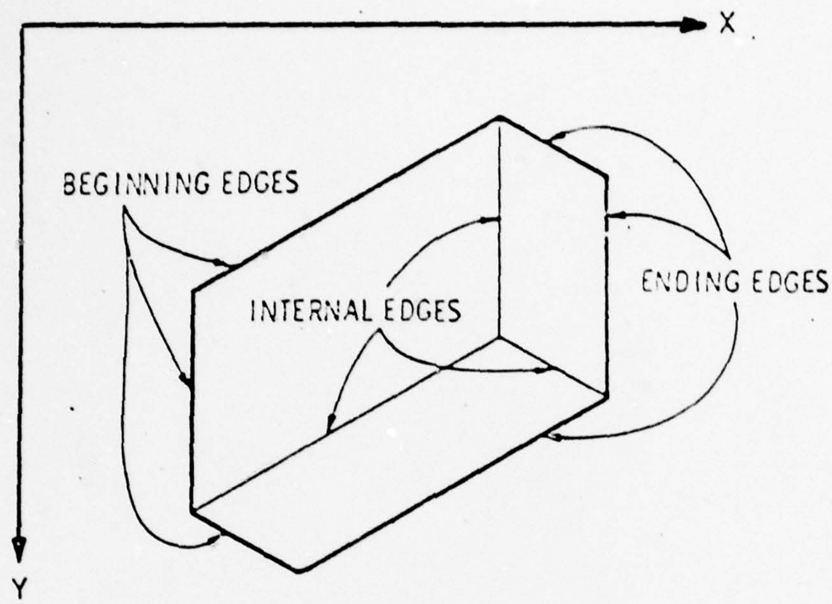


Figure 3.5.1-6 INTERNAL EDGES OF OBJECTS

3.5.2 Scanline Computer

The Scanline Computer hardware is arranged as a pipeline, as shown in Figure 3.5.2-1. All the projected edges of the image are presorted by scanline and stored in the edge list. The update and sort logic retrieves the edges in the order they become active in the image. The sort logic keeps a list of active edges on every scanline, sorted by intersection. The occulting logic examines the sequence of sorted intersections and determines their visibility. The hidden surface problem is solved by an algorithm implemented partially in software and partially in hardware. The objects are ordered by the DIG controller into a priority list. The priority ordering means that if two or more objects overlap in the line-of-sight, the one highest in the list (lowest priority

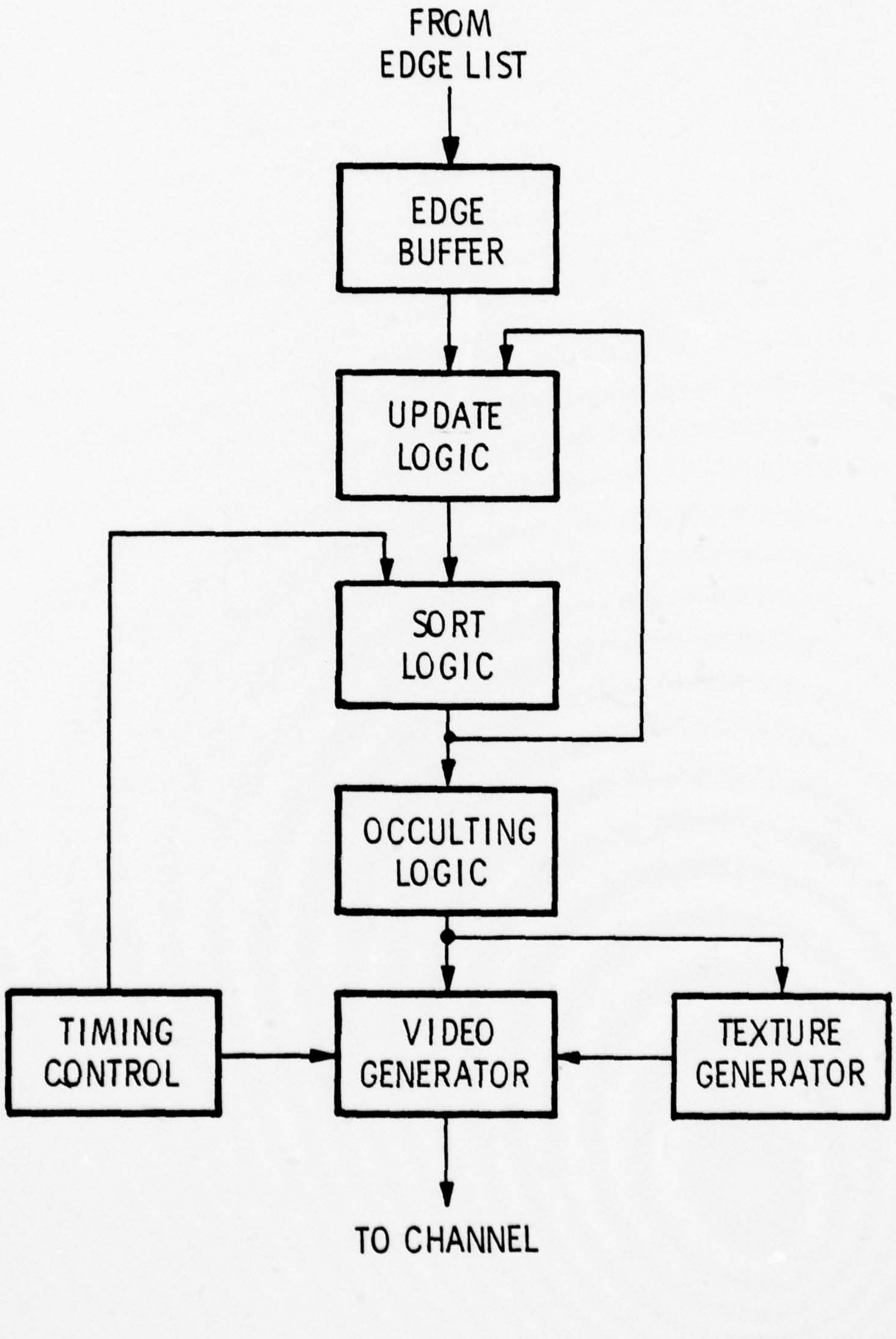


Figure 3.5.2.1 SCANLINE COMPUTER BLOCK DIAGRAM

number) obscures the other ones. Unique ordering of objects is possible as long as certain constraints are satisfied in the modeling of the data base. A sufficient condition is that any three objects be separable by two planes. In the worst case, the condition can be met by breaking up the object into two or more parts.

For some sets of objects the relative priorities can be constant: that is, independent of the viewpoint. Examples of this are stripes on a runway or markings on an aircraft.

The following paragraphs describe in detail the functions of the different segments of the Scanline Computer pipeline.

The Edge Buffer is a 512-word FIFO memory with asynchronous, simultaneous input and output access. Its purpose is to output new edges to the update and sort logic at a high peak rate, with a lower average input rate which is limited by the edge list memory access time. The highest output rate is 512 edges per scanline and the sustained average rate is 8 new edges per micro-second.

The function of the Update Logic is to calculate the intersections of edges with successive scanlines and the intensities and fadings at those intersections. Updating is also used to obtain the true intersection of the edges with the first scanline on which it becomes active.

The Sorting Logic segment of the pipeline keeps in memory the active edges that intersect the scanline. The edges are kept in left-to-right sorted order of the intersections.

The speed of the hardware is such that it can perform sorting of 512 intersections every 38 microseconds. This speed permits the full 512 intersections to be processed for every scanline without the need for large intersection buffer memory. The output of the sort logic is the sequence of sorted intersections.

The Occulting Logic determines the visibility of intersections and calculates the intensity of partially obscured polygons at the point where they become visible.

In order to establish the visibility of intersections, they are examined in a left-to-right sequence on the scanline and a file of "active objects" is generated. When a beginning intersection of an object is encountered, the object becomes active; an ending intersection removes it from the active list. Internal intersections do not change the active status of the object. In a scratch pad memory the intensity and fading at the last encountered intersection of the object is stored, along with the intensity increment, fading increment, and color of the polygons to the right of the intersection.

At each intersection the active object list is examined and updated. The intersection is visible if it belongs to an object whose priority is the highest in the active object list. If the intersection belongs to a beginning or internal edge, the

intensity at that point is determined by the parameters of that edge. When an ending intersection of an unobscured object is encountered, the intensity is determined by the next highest priority object in the active list. The scratch pad memory is addressed to retrieve the last encountered edge parameters of that partially obscured object. For shaded objects, the intensity of the face is calculated at the point it becomes visible, and the fading variable is updated.

The output of the Occulting Logic is a compressed description of the raster scan: a sequence of visible intersections, the intensity and fading at each one, the intensity and fading increment to the right, and the color, if used.

3.5.3 Maximization of Scene Content

It is essential that any scheme intended to maximize scene content begin with the actual data base and data base modeling techniques. It is obviously inefficient and distracting to model areas of the data base with more detail than the system can process in a single frame. When this occurs, the resulting scene detracts from the value of the simulator because the users are distracted by the abrupt elimination and/or introduction of scene content as the system attempts to correct for the overload condition. The frequency and magnitude at which these overloads occur are directly related to the modeling process. An equally frustrating experience for the user is to find sparse data bases or very low detail over entire gaming areas intentionally configured to

avoid any overload conditions.

To minimize the occurrence of both of these extremes, the LINK approach begins by establishing edge budgets to be used in all critical areas of the data base. These budgets are based upon such factors as the features to be modeled, the fidelity necessary for effective use, the scene content of surrounding areas, empirically obtained percentages which indicate how many of the data base modeled edges will become potentially visible at one time, and of course, the capacity of the DIG system. Once these budgets are established, they are given to the modeler as a guideline for the modeling process. The standard which is established for each area of the data base serves as a control mechanism for the lead engineers in reviewing data base models and flagging areas of potential overloads or significant underutilization before the data base is ever even seen. If the expected errors are excessive, modifications are made to the data base at once; otherwise, the areas in which problems may occur are recorded and studied more closely after the image is displayed on the DIG.

This controlled modeling procedure also applies to portions of the gaming area created automatically from the DMA data. In this mode, budgetary edge estimates are made in a manner similar to that described for the manual data base generation procedure. These figures are supplied as input data to the data density analyzer program (DDANAL) and this program automatically performs the monitoring functions which the lead engineer performs

for the manually digitized segments of the data base. The output from DDANAL is fed to other programs which in turn maximize the scene content of specific regions.

Overall, the controlled environment of the entire data base generation procedure leads to the elimination of excessive errors in the edge modeling procedure.

Fine control of the scene content is achieved in real-time through a combination of hardware and software which controls the selection of objects from the data base and which continually monitors and modifies the scene content. In LINK's opinion, maximizing the scene content is not sufficient, by itself, to guarantee an effective viewing environment. The generation of 8000 edge scenes is not difficult. The proper approach is to choose the best 8000 edges subject to the constraints of the operating requirements. To accomplish this end, the real-time functions concentrate on two important aspects: 1) making optimal use of the hardware; and 2) making appropriate adjustments to the scene when an overload occurs, so that the impact on the specific viewing tasks are minimized.

The selection of objects at the appropriate level of detail is straightforward. Each area block of the gaming area contains a list of all the features which were modeled within the boundaries of that block. The features, known as "clusters" within the software system, can each be modeled at up to 12 levels of description. Each level contains two switching distances which

indicate the range at which the more detailed or less detailed version of the cluster should be displayed. The distances are selected so that when a new level is retrieved, the change which occurs in the cluster is not discernible on the screen. The leveling procedure allows objects which are far away from the observer to be displayed with fewer edges than when that same object appears in the foreground. The result is a more effective distribution of edges in the scene. It permits the foreground objects to appear highly detailed throughout an entire mission without overloading the scene complexity.

If, during data base debugging efforts, it is found that the number of edges in the scene at various points in the data base are consistently too low or too high, switching distances and/or the level of detail of various clusters are modified to correct the error.

Once the data is retrieved from mass storage it must be processed through the DIG. Failure to perform some field-of-view calculations results in wasting DIG processing on an object which will never be seen. This processing could be done solely in the visual computer, but that is a very time-consuming procedure. Therefore, The LINK system incorporates a feature known as "artificial objects". These objects are cubes, composed of 12 edges, which completely enclose a cluster of actual object.

To determine whether or not the cluster should be processed for a given field-of-view, the artifical object is sent to the

DIG and a visibility code is returned to the CPU. If the artificial object is totally out-of-view, then the cluster need not be processed. Thus the processing of a simple 12-edge cube saves the processing of hundreds of edges which will ultimately not be displayed. If the visibility code indicates that the object is totally or partially in view, then the cluster is processed and there is a high degree of certainty that at least part of the cluster will be seen on the display.

The use of artificial objects in this fashion for all clusters in the data base results in more efficient utilization of the DIG hardware.

However, leveling of detail and the use of artificial objects are still not sufficient by themselves to achieve maximum scene content. In many cases the presence of fine detail in the scene is the single most significant contributor to proper speed, acceleration, or distance cues. This fine detail and texture appears very gradually in the scene, but appears at close ranges where normal software retrieval of newer levels of description would result in the abrupt appearance of such detail. If the more detailed levels were brought into the system much earlier than necessary, the fine detail would be processed but would not be resolvable on the scene; hence some edge processing would be wasted.

To achieve the graceful introduction of detail into the scene, LINK has incorporated two special hardware features known as resolvability codes and subpriorities. Briefly, they allow a modeler to specify up to seven hardware levels of description for any particular face. Each succeeding level contains a bit more detail that its predecessor. The choice of which level to display is determined through the use of resolvability codes. These codes specify a minimum size at which the level of detail should appear on the screen. If, when the detail is projected onto the image plane, it would be smaller than the specified resolvability code, then that level of detail is not processed through the system.

The utilization of levels of detail and resolvability codes within the DIG allows the hardware to eliminate, early in the pipeline logic, those edges which, because of their size, would not be discernible on the screen.

The result of all of these techniques is that the system processes the appropriate level of description of clusters which are known to be in the field-of-view, and furthermore, that all edges which are too small to be resolved are automatically excluded from processing. Thus the user is assured that the final 8000-edge scene represents the most effective utilization of the DIG hardware.

However, even with the controlled modeling and judicious selection of detail, there is still a chance that the edge capacity of

the system will be exceeded. When this occurs, a dynamic response must correct the error with minimum impact on the scene.

When an overload occurs in the LINK system, it is not the result of a gross error in the number of edges. These excessive overloads are tuned out of the system by means of the controlled data base generation procedures and careful selection of switching distances and levels of detail. Also, they will not appear regularly at specific points in the data base because these errors are also corrected off-line. They will result from some unique set of circumstances which may occur and then disappear over a few seconds. While the overload occurs, the system must reduce the edge processing load on the system in some graceful fashion.

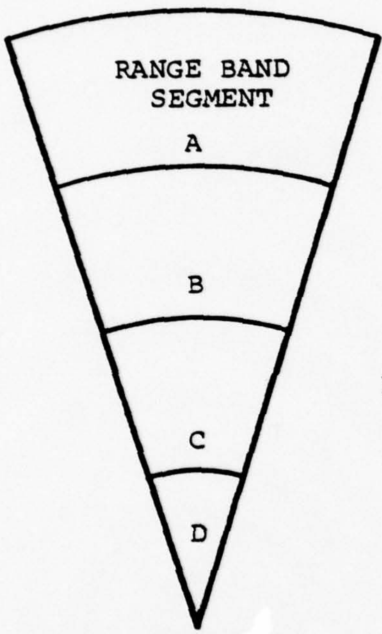
This reduction is accomplished by rescaling the resolvability code testing in the hardware so that that the detail which would normally be brought into the scene is excluded from processing. The result is that the very smallest detail faces are dropped out of the scene and restored gradually at a later time when the causes of the overload are no longer present.

It is important to note that the change in the resolvability code processing may only affect a few edges, but this is sufficient for the majority of cases of this type.

When this technique is not successful, a more serious problem exists. In order to allow operation to continue, the software will take less graceful measures and remove some of the farthest clusters from processing altogether. This much less common error

will be recorded and then examined and corrected off-line.

The edge budget statistics of a typical maximized displayed image will be approximately as follows:



Range Band	Outer Limit (Miles) of Range Band at Nominal Altitude of 1,000 ft.
A	25
B	12.5
C	6.25
D	3.13

Edge Distributions:

Range Band	Object Edges Other than Texture	Edges Used for Texture Effects	Totals
A	2,000	0	2,000
B	1,600	400	2,000
C	800	1,200	2,000
D	300	1,700	2,000
	4,700	3,300	8,000

Figure 3.5.3-1 EDGE DENSITY VERSUS RANGE

3.5.4 Scene Enhancements

In addition to the development of the large edge processing capacity of the DIG system, LINK has included in its system design a number of optional special features which enhance the quality of the generated scenes. Among these features are:

1. Curved Surface Shading. Although all objects in the LINK DIG system are composed of planar faces, any flat-faced object can be made to appear to be rounded through the application of smooth shading. In this technique, the first step is the construction of vectors at right angles to the flat faces. These "normal vectors", however, apply to whole faces, so that the illumination changes abruptly as the scanline crosses from one face to the next. To make the variation smooth, the normal vectors for a group of adjacent faces are averaged to produce a vector that points in a direction median to the others. This vector is placed at the vertex where those faces join one another; known as a "vertex normal", it is part of the data base. During every frame in real-time, the angle between this vector and the line to the sun is computed, and a brightness calculated from the cosine of the angle -- except that the brightness is assigned to the vertex rather than a whole face.

   The next step occurs after the projection of the vertices to the picture plane. As a scanline passes through the picture, cutting through a display edge, a brightness value

at the point where it cuts is computed by interpolating between the brightness values at the upper vertex and the lower vertex of that edge. The same is done for the other edge defining that same face on this scanline; then the system interpolates between those two brightness values along the scanline. The result is a smooth shading which gives the object a rounded appearance, and which realistically changes as the angle of the object with the sun changes.

Smooth shading can be applied to flat objects to create special effects. For instance, an extremely realistic representation of a nimbus cloud layer can be achieved by arranging a number of rectangular faces on a flat surface and applying smooth shading. The vertex normal vectors are arranged to point in more or less random directions; under the influence of the sun shading computations, then, the faces take on an undulating grey shade which faithfully duplicates the appearance of this type of clouds.

In another application, smooth shading can be applied to mountainsides to create the illusion of depressions or gullies, without the need to sacrifice objects in the data base -- addition of this type of "gully" does not affect the original convexity of the mountain, so it continues to count as only one object.

2. Dynamic Illumination. The DIG permits the dynamic modification of sun angle, intensity and diffuse illumination each

AD-A059 536 SINGER CO SUNNYVALE CA LINK DIV F/G 15/4

ADVANCED DIGITAL SIMULATOR SYSTEM (ADSS).(U)

JUL 78 DAAK70-77-C-0175

UNCLASSIFIED NL

2 OF 2

AD A059 536

END

DATE FILMED

12-78

DDC

frame. This permits effective simulation of various lighting conditions: bright daylight, dusk and nighttime conditions, both for visible and IR wavelengths.

Sun shading, or illumination, is accomplished for all faces by computing, for each one, a vector that points outward at right angles from the face. The angle between this vector and the sun illumination vector is then found. The brightness of the face is adjusted so that the smaller the angle (the more the vector points directly at the sun) the brighter the face will be. A cosine function of the angle has been found to be useful and easily derived. The diffuse illumination factor is added to this cosine function.

3. Texture. Surface texture is very important for providing attitude and motion cues to the pilot. However, if texture were simulated with edges, the number required would be astronomical. The texture algorithm recently developed by LINK uses no intersection capacity of the system. Some of the features of this scheme are as follows:

1. It can be implemented in real-time hardware.
2. The planar surfaces on which texture is superimposed applied to terrain faces, both 2-D and 3-D, and may have any orientation on a face.
3. The texture can be superimposed on smooth shaded objects and can be contiguous across edges.
4. The texture patterns are not repetitious. They depict the natural characteristics of the surfaces onto which they are superimposed.
5. The modeler has complete control over the kind of texture he may want to superimpose on the surfaces.
6. The modeler can specify different kinds of texture on different surfaces.
7. The textured surfaces are constant in time, are in correct perspective, and are occulted as appropriate.
8. The texture is spatially fixed to the faces and therefore will change proportionally in display size in accordance with the perspective processing of the faces.

In order to explain the approach Link has developed, it is necessary to image a big texture look-up table covering the ground plane of the whole gaming area. The texture on the ground plane is orthographically projected onto the designated surfaces in the data base. In the non-horizontal case, the texture will appear to be "stretched" in the proportion to the inclination of the surface from the horizontal (see Figure 3.5.4-1). In all cases it is only necessary to obtain the X_e and Y_e coordinates of each displayed point on the surface. The coordinates are then used to look up the texture intensity from the texture look-up table. The resulting texture intensity is then superimposed on the intrinsic intensity of that point.

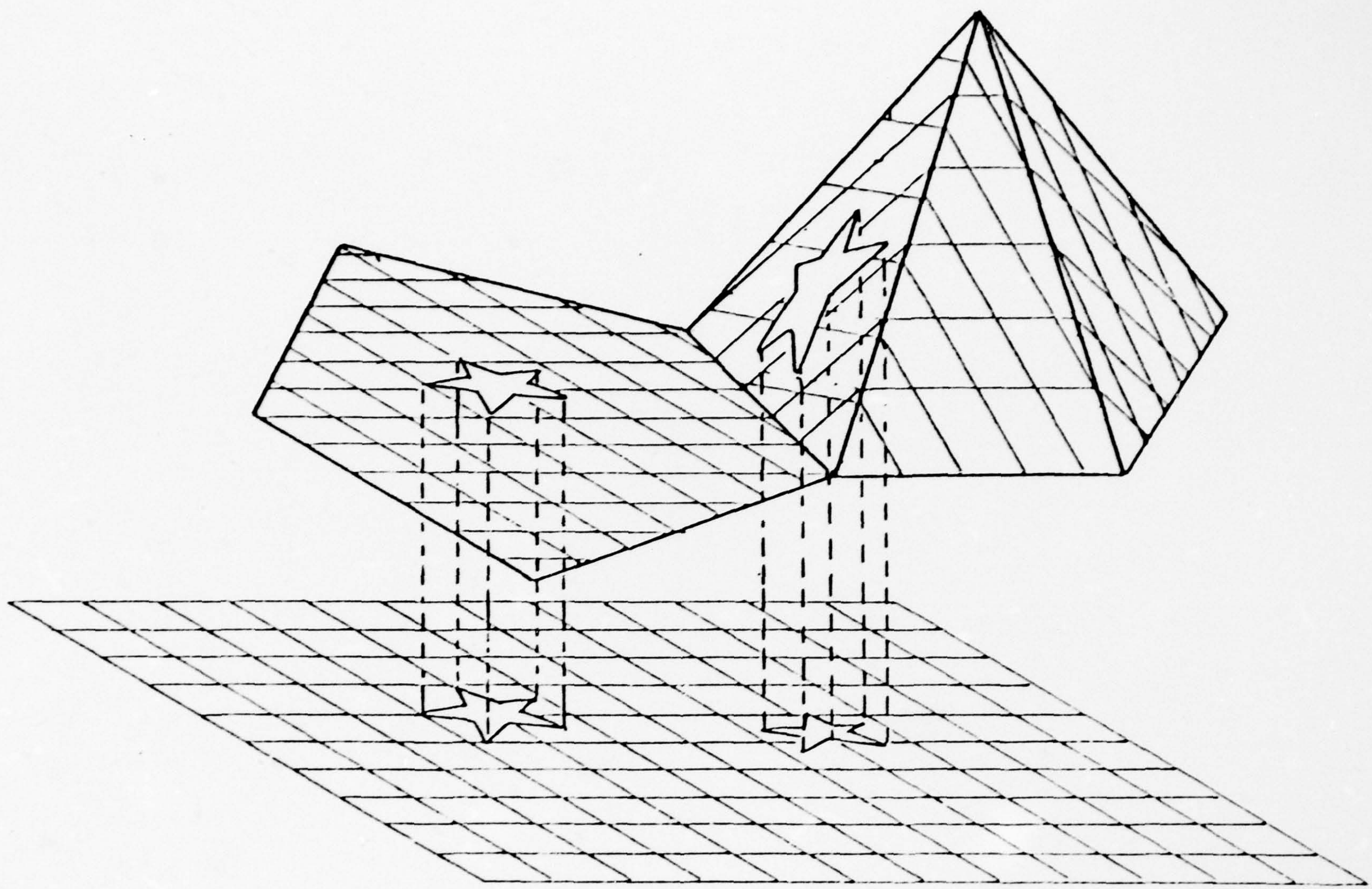


FIGURE 3.5.4-1 TEXTURE ON HORIZONTAL AND NONHORIZONTAL SURFACES

Since the X_e and Y_e coordinates are calculated only for the displayed points, time is not lost in calculating X_e and Y_e for points on an occulted surface. This is important, since it guarantees that textured surfaces are constant in time, and in correct perspective.

Evidently, even at low resolution, the size of the PROM or RAM required to form a texture look-up table covering a 10 X 10 nm gaming area would be prohibitive. For example, let the resolution be such that a texture element (a cell in the look-up table) is of size 4' x 4'; let each texture element be digitized to 64 intensity levels using 6 bits, the PROM/RAM size required to cover the 10 x 10 nm gaming area will be approximately 1.3 billion bits.

We utilize a "Tile approach" which uses PROM's of reasonable sizes. A tile consists of a reasonable number of texture elements; for example, a tile consisting of 64 X 64 texture elements, each of size 4' X 4', will cover an area of 256' X 256' and require a PROM/RAM size of 4K X 6 bits. (The factor 6 comes from the assumption of 64 intensity levels for each texture element). The simplest way to form a texture look-up table covering the whole gaming area from one single tile is to repeat the same tile over the entire texture look-up table. However, this will not be acceptable since the texture pattern will look repetitious.

This approach is to generate the tile in such a way that the intensities along the four sides are matched, so two tiles in any orientation could be laid side by side without sharp transitions. The eight orientations can be obtained from the PROM/RAM of the single tile by addressing the port A & B of the PROM/RAM as shown in Table 3.5.4-1. The bar indicates 2's complement.

	A	B
1	X_e	Y_e
2	Y_e	X_e
3	$\overline{X}_e$	$\overline{Y}_e$
4	$\overline{Y}_e$	$\overline{X}_e$
5	$\overline{X}_e$	Y_e
6	Y_e	$\overline{X}_e$
7	X_e	$\overline{Y}_e$
8	$\overline{Y}_e$	X_e

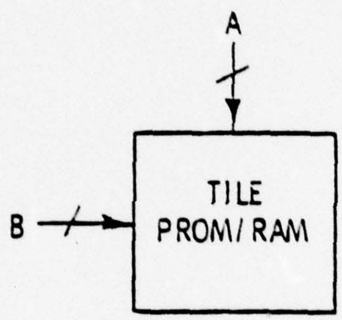


TABLE 3.5.4-1

The 8 orientations obtained from the same tile PROM/RAM will be scattered over the texture look-up table pseudo-randomly, in accordance with the natural characteristics of the surfaces.

Figure 3.5.4-2 shows the picture of a tile. Figure 3.5.4-3 shows one possible arrangement of the tiles. To prevent scintillation, texture must gradually fade away as the texture element becomes too small on the screen. To achieve this, the approximated minimum width (in terms of the number of scanline widths) of the texture element in the neighborhood of the displayed point is obtained, and when the value becomes small the texture elements are faded out.

This texture generation approach can be easily modified to incorporate several levels of detail. All that is necessary is to define a few more tiles at different resolutions. The texture intensities of the different levels are then faded, and summed to give a resultant texture intensity. Figure 3.5.4-4 illustrates three levels of detail whose texture element sizes are 1' x 1', 16' x 16' and 64' x 64' respectively. This figure also illustrates texture fading with distance.

4. <u>Edge Smoothing and Antiscintillation Circuits</u>.

The requirements are estimated to be minimal for EVS simulation, where image resolution is already degraded by use of EVS Video Effects Generation, etc.

FIGURE 3.5.4-2 PHOTOGRAPH OF TILE

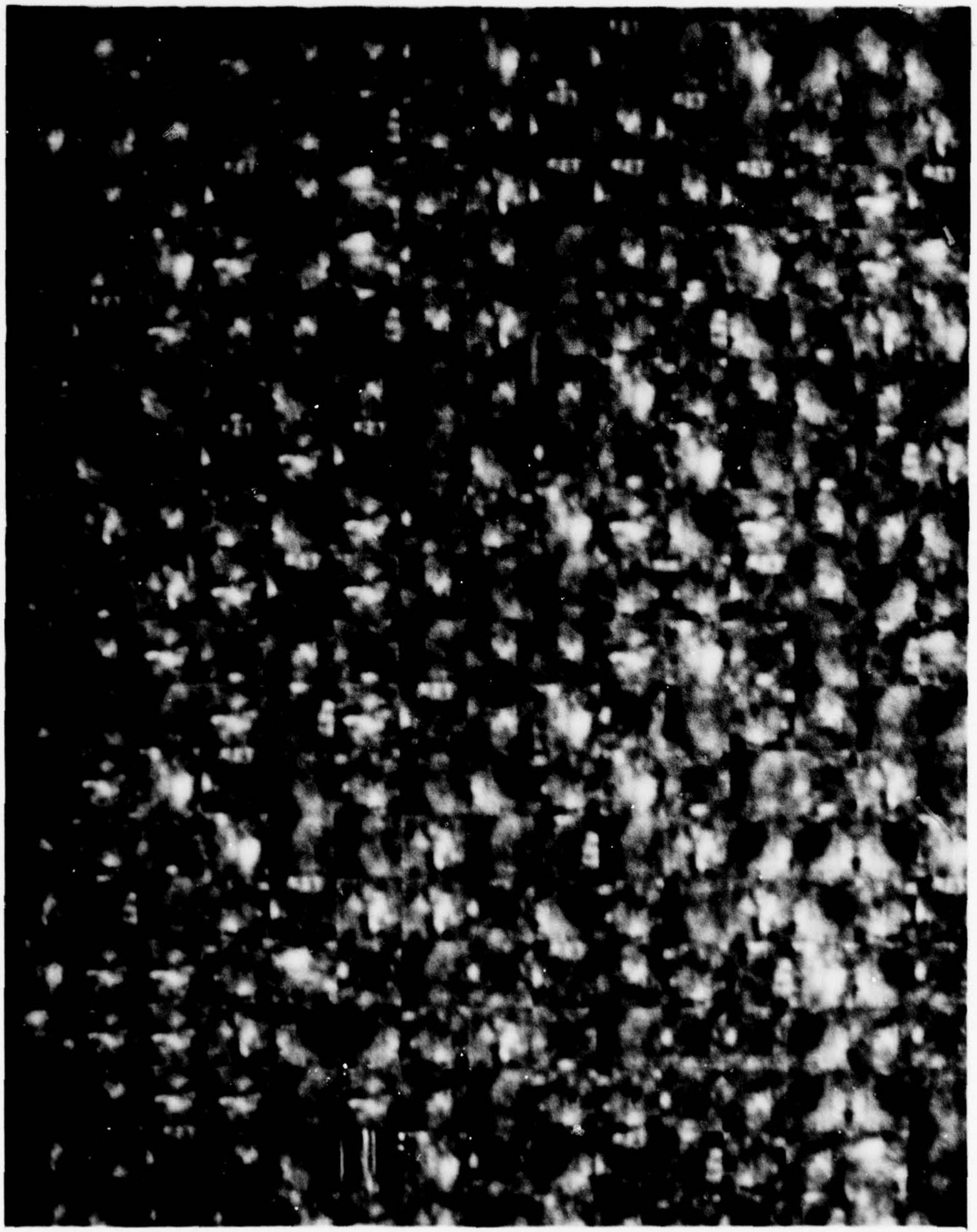


FIGURE 3.5.4-3 POSSIBLE TILE ARRANGEMENT

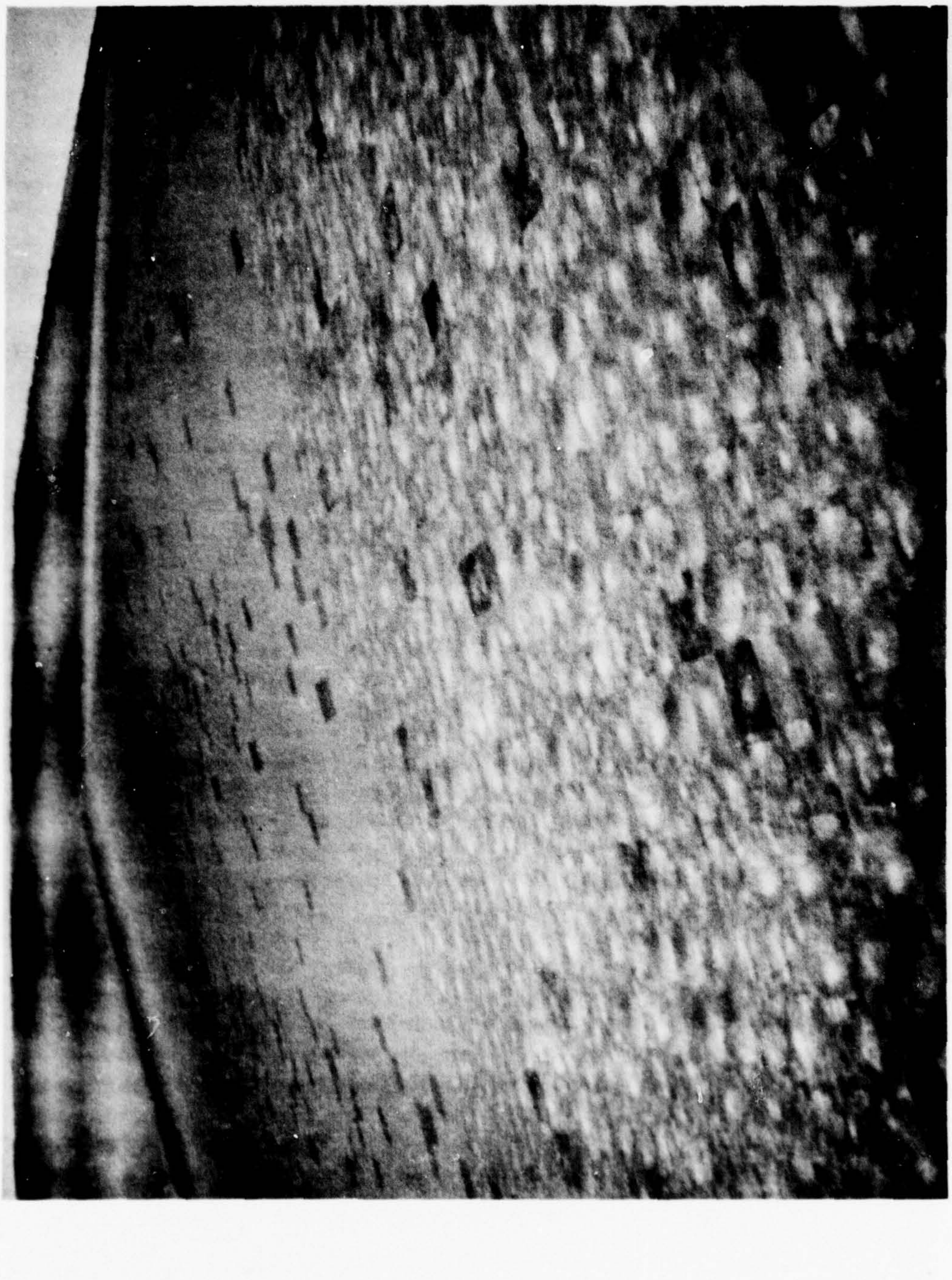


FIGURE 3.5.4-4 EMULATION OF THE TEXTURE ALGORITHM BY THE LINK IMAGE RESEARCH LABORATORY

5. Updating Rate

The cost of providing scene updating rates of 60 times per second rather than 30 times a second is disproportionately high. There doesn't seem to be need for the high update rates in ADSS because some image degradation at high platform slew rates should not detract from the evaluation of sensors to nearly the same degree that it affects pilot training missions.

6. Level of Abstraction

At all times it should be kept in mind that the digital sensor simulator provides an abstract image. The capability of the digital image processing hardware determines the level of abstraction. It is the task of the base modeller to use the maximum capability of the processing hardware to make effective scenes. Since creating digital scenes is a relatively new art extensive investigation will be required by both vendor and user to determine what scene objects and what levels of detail are required for the missions.

3.5.5 EVS VIDEO EFFECTS GENERATOR

One of the most critical aspects of EVS simulation is that of providing realistic simulation of the non-ideal characteristics of the EVS sensors. Several characteristics which commonly affect the output of electro-optical image sensors are:

1. Signal-to-noise ratio
2. Resolution and/or focus
3. Threshold and limiting
4. Nonlinearity
5. Blooming
6. Overload
7. Automatic gain control or iris control
8. Slow image decay
9. Sensor element mismatch

To provide these effects, an EVS video effects generator can be used. Figure 3.5.5-1 is a block diagram of one type of video effects generator.

The system employs three primary video paths. First is the forward path which is used for inserting those effects which may be added by simple electronic signal processing means. These effects include signal-to-noise ratio, threshold, limiting, nonlinearity, gain control, and sensor element mismatch. The second path provides a forward video path through a display and camera pickup. This path provides for inclusion of the effects of resolution

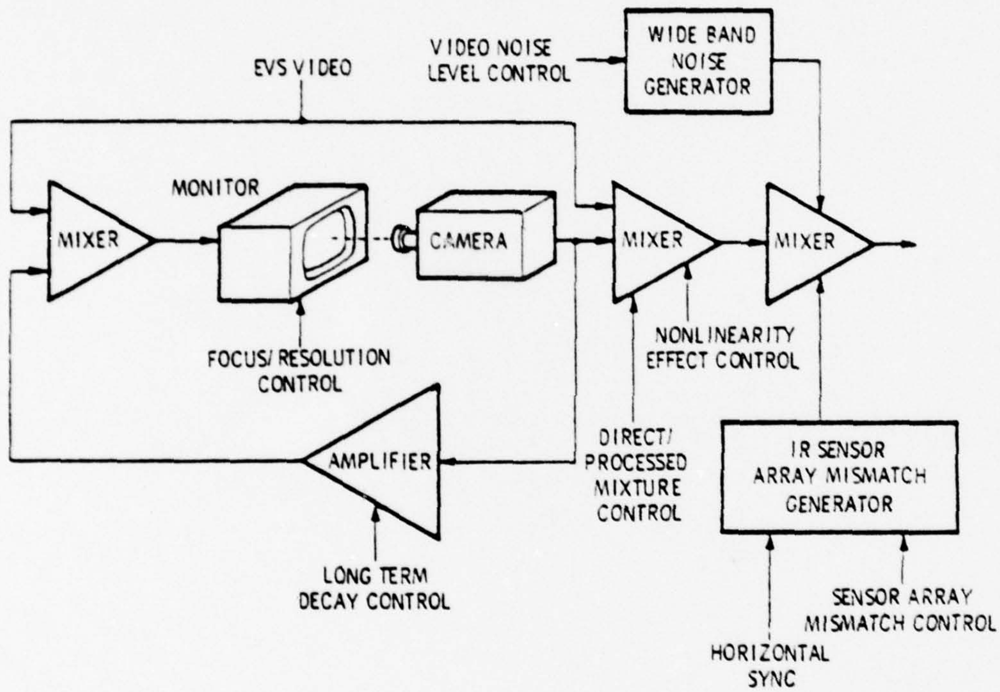


Fig. 3.5.5-1 Video Effects Generator

and/or focus, blooming, and overload. The direct electronic forward path bypasses the display/camera chain in order to provide nondegraded video at the output which is then mixed with the camera output. The third path provides video feedback to the display from the camera. This path provides for simulation of slow image decay effects.

3.5.6 ATMOSPHERIC ATTENTUATION

Visibility will be controllable at the instructor's station for FLIR/LLLTV. The effect of atmospheric attenuation will be to reduce the contrast of scene elements as a function of range. All intensities will tend toward a background intensity C_b which will be manually settable for FLIR and LLLTV. The fading function will be as follows:

$$F = e^{-R/K(VR)}$$

where:

R = distance to scene element

VR = visability range

K = empirical constant

The faded intensity value in terms of the fading function will be:

$$C_F = C \cdot F + C_b\,(1-F)$$

where:

C_F = faded intensity

C = unfaded intensity

C_b = background intensity

F = fading function

The fading function is computed continuously over the surfaces of objects in the scene. This method of fading generation is essential to the realistic simulation of rolling or mountainous terrain viewed at low altitude. Other methods compute fading on a two-dimensional ground plane or compute it on an object-by-object basis. These appear very unrealistic when applied to large objects extending above the ground plane.

3.5.7 SUN POSITION EFFECTS

The effect of sun angle is an important consideration in the realistic simulation of IR imagery.

For processing FLIR intensities, the image processor employs a table look-up method to translate IR descriptor codes contained in the on-line DDB into IR image intensity. The contents of the table will be loaded as a function of time of day to permit real-time simulation of diurnal effects. This method is necessary because the diurnal effects cannot be completely determined as a function of sun angle. The IR descriptors in the on-line DDB will be assigned according to such considerations as surface or structure material, and type of structure or terrain, as can be determined from the planimetry file of the DMA data. For each descriptor code, data is stored in the table to relate the IR intensity for horizontal surfaces, surfaces facing toward the sun, and surfaces facing away from the sun. The general-purpose processor will contain data describing curves of these three variables as a function of time of day for each IR descriptor

code. The data corresponding to the particular time of day is then extracted from these curves and loaded in the IR intensity table located in the special-purpose image processor hardware. This method provides not only for diurnal effects but also allows for programming of seasonal changes, and latitude effects through changes to the curves modeled in the general-purpose computer software.

The special-purpose hardware will then compute IR intensity as a function of the tabular value and the sun angle. The values for horizontal surfaces may be used directly. For other surfaces the surface normal vector will be tested against sun vector to determine if the surface faces the sun. If it does not, then the value tabluated for surfaces facing away from the sun will be used directly. If the surface is found to be facing the sun, then an interpolation will be made between the sun illuminated value and the non-sun illuminated value as a function of the angle between the sun vector and the surface normal. Figure 3.5.7-1 is an example of the curves used for simulation of diurnal effects for the Las Vegas area of the DMA DDB.

3.5.8 MOVING OBJECT MODELS

Several simultaneous moving object models must be provided for use in display of realistic mission scenarios in ADSS. These objects are typically moving vehicles, together with special effects affecting their visibility.

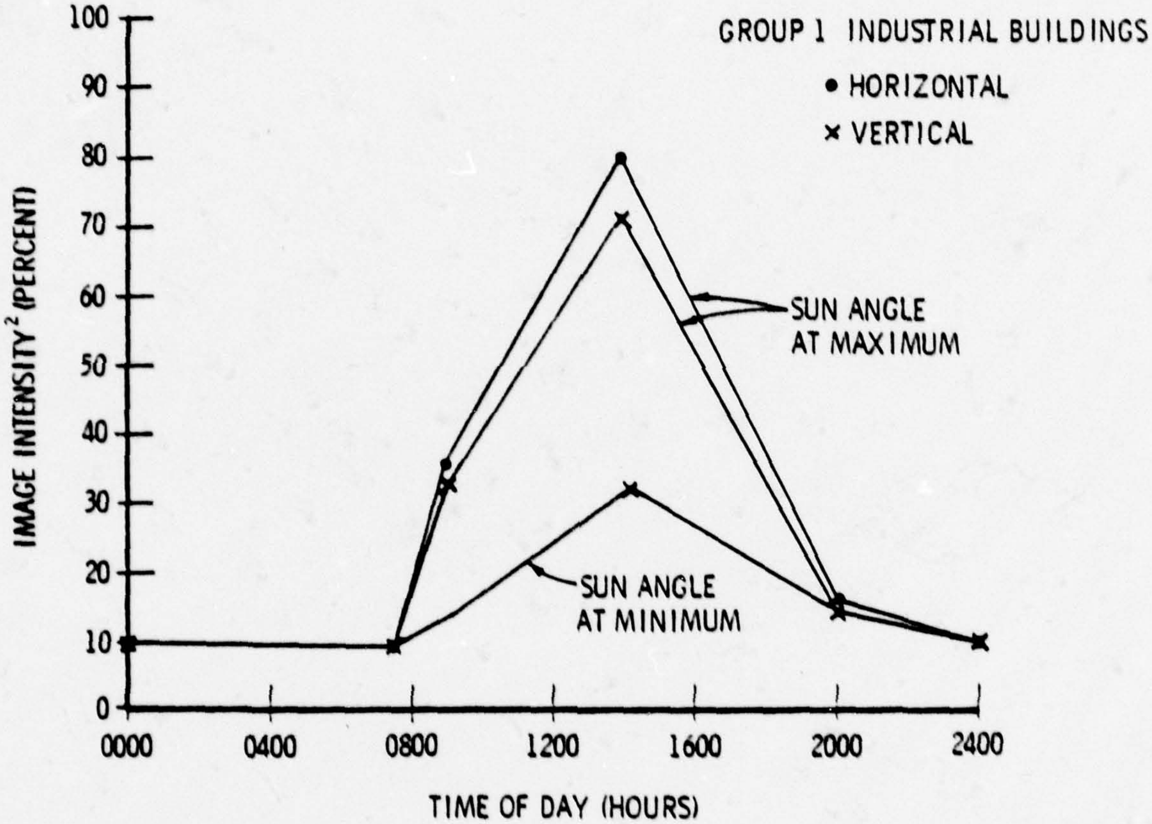


Figure 3.5.7-1 NORMALIZED IMAGE INTENSITY CURVES FOR GROUP 1

Generally, moving vehicles do not require very detailed levels of description. Their size and distance from the ownship are such that coarse representations sufficient to permit recognition of the vehicle are all that is required. However, other vehicles may at times come very close to the ownship, and in such cases additional detail is necessary. For example, coarse descriptions are not capable of providing adequate visual cues for ranging and closure rate at close range. For this reason, the moving features are modeled in various levels of detail, and different descriptions of the vehicle are displayed depending on the distance to the object.

The data base modeling of the objects will be based upon drawings, specifications, and photos of the vehicle. In addition to the basic structure, the models will include all insignia and lights necessary for effective testing. Each type of vehicle is modeled only once (although it may have numerous levels of description) but can be used repeatedly in real-time if more than one vehicle of that type is selected. The visual software will take care of dynamically retrieving the appropriate level of description for each of the vehicles. The software will also insure that the displayed image of each vehicle will properly respond to the position and attitude calculations for the vehicle, occlude and be occluded by other objects in the scene, and properly reflect the effects of sun illumination on the vehicles.

For example, if it is assumed a mission scenario is developed for an exercise covering a small gaming area, possibly 100 nm^2, the imaging effects might be as follows:

Scene Content:

- Rural scenes
- Village scenes
- Open or wooded terrain
- Image density: adequate for camouflage
- Level and hill terrain mixture
- Varied routes through terrain: to avoid memorization
- Adequate detail for range estimation
- Ground texture cues to indicate softness or stability
- Textured cues to slope of terrain
- 10 to 25 target images: moveable and recognizable types (nominal)

Target Image Characteristics:

- Accuracy of location and attitude: with sufficient detail for identification
- Realistic attitudes in terrain
- Occultation by foreground detail
- Occultation of background detail
- Correct "hit" effects
- Precise correlation with fire control indications and with designator indications.

Special Effects:

- Illumination: day and night, also self-emissions of infrared
- Weather: haze, fog, rain, heat shimmer, snow/ice
- Threat effects: smoke, dust, glint

Weapon Effects:

- Tracers, main gun and machine guns
- Tracer penetration and richochet effects (possibly)
- Burst on target
- Firing flash: threat and ownship
- Missile and Missile Plumes
- Smoke grenades and target obscuring effects

A moving object need not be a single object, it can be a group of objects with common motions, such as a convoy of trucks, a formation of aircraft, etc. With both the fixed scene content objects and the moving objects, a list priority approach is used for solving both the hidden surface problem due to object position and attitude and the occulted surface problem due to position changes of moving objects. Relative priorities are computed by the digital control computer in terms of an absolute priority number which is determined for each convex object to be generated. The priority numbers must be such that if two objects overlap when projected on the image plane, the object with the lower priority number occludes the other object. Two overlapping objects with the same priority number essentially cancel each other in the region of overlap. This is the reason for requiring convex objects, since it is impossible for such objects to over-

lap themselves when projected onto the image plane (provided back-facing polygons are ignored).

The hidden surface problem is reduced to the assignment of priority numbers to convex objects. Since, in general these priority numbers will vary depending on the instantaneous observer position, they cannot be incorporated as intrinsic properties of modeled objects at the time of data base generation but must be determined based on the interrelationships between the observer and the entire set of displayed objects.

Where relative priorities among groups of objects have not been previously specified several approaches are used to determine them on a dynamic basis. The first of these approaches involves the use of visibility separating planes which divide the group of objects into subgroups in a binary, recursive manner until each subgroup consists of a single object. This process is illustrated in Figure 3.5.8-1 for the set of objects (A,B,C,D,E, F). Plane 1 is seen to divide this set into two subsets, (A,B,C, D) and E,F). The latter subset is again divided into (E) and (F) by Plane 2, and so on. The entire process is conveniently represented by a binary tree structure, as indicated in Figure 3.5.8-2, wherein the circled nodes represent the separating planes, and the square terminal nodes represent the objects. The (+) and (-) branches emanating from each plane node correspond to the (+) and (-) signs associated with each plane in Figure 3.5.8-1.

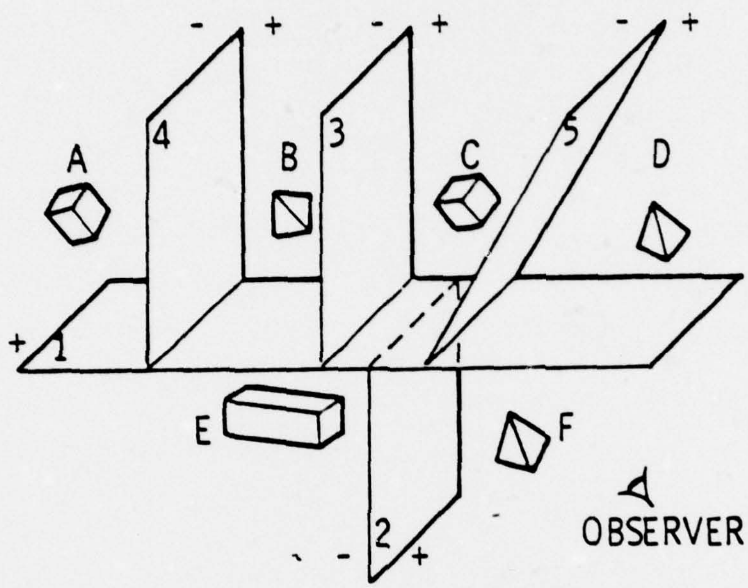


Figure 3.5.8-1 SEPARATING PLANES

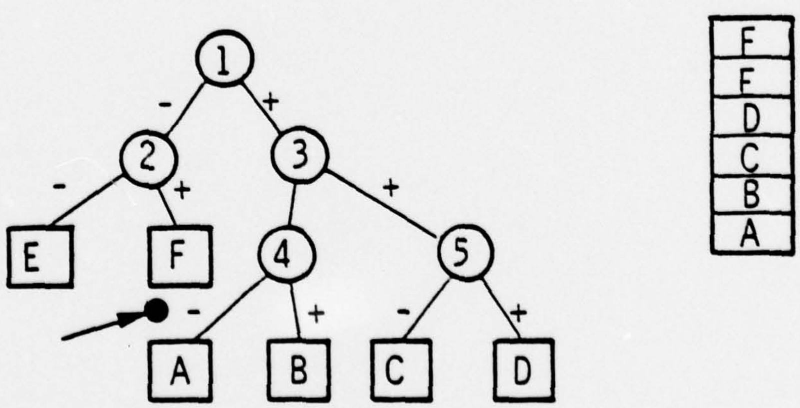


Figure 3.5.8-2 SEPARATING PLANE TREE NODE PRIORITY LIST

The relative object priorities are determined in real-time by searching the separating plane tree, which is generated off-line by the data base compiler. The search consists of a prefix walk of the tree, with the selection of which node branch to follow first being made on the basis of whether the observer position is on the (+) or (-) side of the plane. As an example, for an observer positioned as in Figure 3.5.8-1, the process would start at the root node (Plane 1) and determine that the observer is on the (-) side of this plane. It would then check Plane 2, to find that the observer is on the (+) side. Since this branch terminates in an object (F), this object is assigned the highest priority, and the process now backtracks through the tree, checking all unexamined nodes on a last-in, first-out (LIFO) basis. Thus, it finds that object E has the next highest priority, and then backs up to examine the (+) branch of plane node 1. The process continues until the unexamined nodes have been exhausted. Thus the priority list to the right of Figure 3.5. 8-2 is obtained. This search procedure has a compact program implementation using LIFO stacks.

This approach is used to perform calculations for objects in moving clusters as well. Moving clusters are tested against the separating planes for fixed objects to determine which subgroup they are in, until the subgroup consists of one object. Then a separating plane at the moving object position establishes the priority.

As can be seen, this process is fairly complicated, and therefore limits the maximum number of moving objects which can be handled simultaneously. The object image description can be quite readily called up from memory and the level of detail selected for object distance, but the occultation effects by and on other objects requires evaluation of priorities. Since the objects move relatively slowly, the examination of priorities for each frame need only be a local one, but still can be quite time-consuming, for example a down-the-road view of a column of moving vehicles, which has multiple occultations plus possibly smoke obscuration, glint effects, etc.

3.6 MODELING THE GAMING AREA

There are two basic methods of modeling the DIG gaming area. These are manual modeling through the use of the Data Base Modeling Equipment provided, and automatic data base generation thru computer transformation of DMAAC data.

3.6.1 Manual Modeling

An independent capability is provided in order to create gaming areas off-line as a customer in-house function. It utilizes the Interdata 8/32 computer, moving-head disk, digitizing equipment, control means, and a special line-drawing display for data base verification. It includes equipment capable of providing hard-copy documentation, not only of the line drawings, but of full printed annotation for them; thus producing a formal book usable for backup and as reference for all future changes, without need

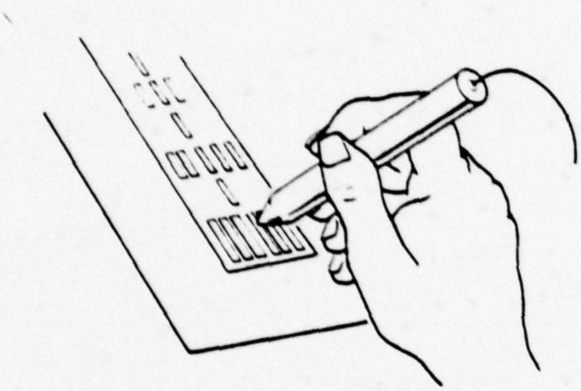
for manual work.

The system includes all hardware, software, and library data base features necessary to allow customer personnel to locate objects, and generate particular scenes edge by edge. The modeling equipment allows the generation of any shape of any desired emissivity on a piecewise, or edgewise, basis. It allows rotation, translation, and profile modification of given gaming area features. The output of the gaming area modeling equipment is a magnetic tape file suitable for loading and storage on the gaming area rapid access storage/retrieval system. The equipment models on a real-time basis -- that is, as each part of a gaming area is modeled, it can be displayed on the CRT. The perspective of the CRT is variable under the operator's control. The modeling equipment is sufficiently flexible and detailed to allow a trained and knowledgeable operator to create a representative gaming area through the selection of gaming area features from a feature library, and their placement in the gaming area through digitization from a map or maplike drawing.

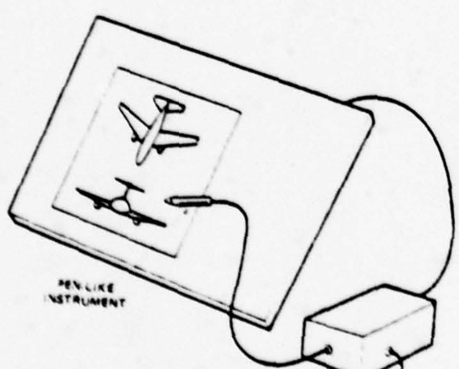
The DIG modeling equipment - hardware and software - consists of a number of powerful tools which provide the customer with a practical, in-house capability.

In an effort to make the environment modeling process as direct and as natural as possible, LINK has chosen to build its modeling system around a wire-mesh XY digitizer table and pen. With this device, digitizing is done with a convenient hand-held instru-

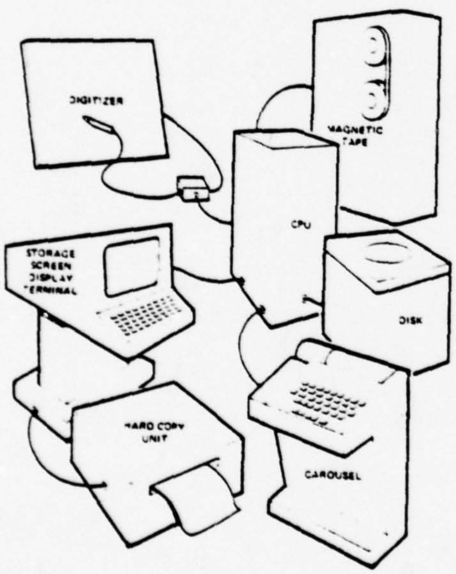
ment similar to a pen. It is touched directly to points on a drawing taped to the flat surfaced digitizer table, and the locations of these points are then automatically communicated to the computer and the data base, without need for paper forms and punched cards. Convenient controls for the digitizing process are arranged directly on the digitizer surface, so that control functions are performed with the same pen-like instrument, touched to "buttons" in the control area. Since there is no need to put the instrument down, the modeler can proceed rapidly without becoming fatigued.



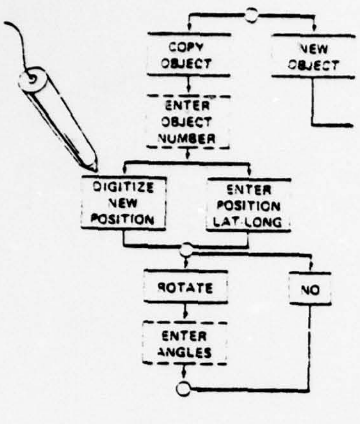
Typically, for three-dimensional objects, drawings consisting of both a top view and a front view of the object would be placed on the digitizing surface; touching the corresponding point on both views will communicate all three coordinates (X, Y, and Z) to the computer, which for limited areas can be thought of as the latitude, longitude, and elevation of points on the object.



Coordinates picked off on the digitizer are sent to the computer, where extensive special software rescales them to data base units, and stores them on a moving-head disk in an easily updatable format. Line drawings of what has been digitized, along with printed annotations, appear on a storage-screen display terminal, which is also driven by the computer. A hardcopy unit connected to the display terminal makes paper copies of any desired material on the screen. A carousel simplifies communication with the CPU, and a magnetic tape unit provides back-up storage for the disk and allows its data to be sent to remote locations.

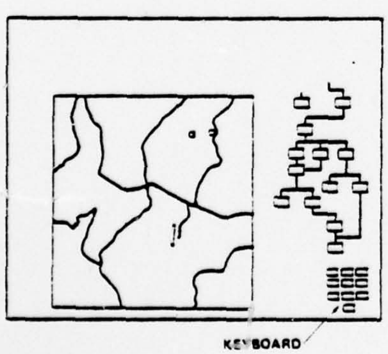


As noted above, control of the digitizing process is accomplished by touching the digitizer pen to "buttons" directly on the same surface as the drawings of objects. These "buttons" are nothing more than pictures on pieces of paper, affixed permanently to the surface; each button is a half-inch square box, the location of which is known to the software. Control of the process is greatly simplified by arranging these buttons in a flow chart, the branches of which represent the options open to the modeler.

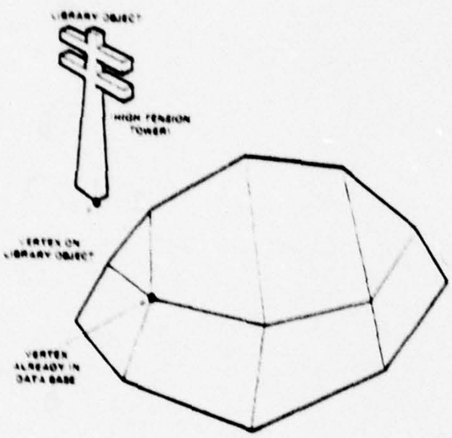


The entire process is laid out before him graphically in the form of this flow chart, and to guide the process through various options, he has only to select paths by touching buttons actually shown in those paths.

Typical of the functions that are controllable through the flow chart are some that greatly increase the convenience with which the digitization can be carried out. For instance, the modeler may want to work at a scale which shows an entire gaming area at once, yet occasionally position some *object* (copies from the library of objects) to an accuracy much finer than such a course scale would allow. An option is provided in which the modeler can enter the coordinates of a specific point numerically - on a keyboard which is also simply a group of paper "buttons" at the foot of the chart. He thus avoids having to construct a special drawing at an expanded scale. Many such special functions are provided.

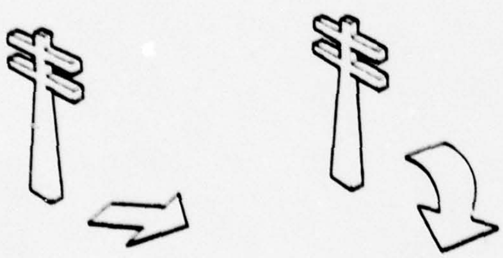


Next, the modeler can refer to the automatically produced documentation for the library objects, and select one for placement in the gaming area. Its identifying number (assigned by the



system) will be shown in the documentation, and entering that number will recall the object from the disk. If no modification is needed, it can be placed in the gaming area directly by digitizing on the map or entering an X, Y, and Z position.

Still another method involves moving the library object from one position to another by deltas in X, Y, and Z, which would be useful after it has been put into the gaming area, to nudge it into place if necessary.



The library object can also be rotated, usually in azimuth, but in all three angles if necessary.

3.6.2 Automatic Modeling

The limited edge processing capacity of the DIG requires that the DMA data be greatly compressed before processing by the DIG. When 8000 potentially visible edges are spread out over the FOV, at an altitude of 1000 ft. and out to a range of 25 nm, the average edge density in the scene is considerably less than that found in the coarsest DMA elevation data. Consequently,

the shape, location, and detail of significant landmarks may be different from those in the real world, or in the DMAAC data. Critical differences will require modification through manual digitizing.

Although DMAAC data furnishes high-density information, the accuracy of the data, especially after it is transformed into a visual data base, may not be sufficient in those areas of the data base in which high detail is required. These critical regions can be corrected by manual digitizing. In addition, much of the cultural information in the DMA data is insufficient to generate realistic sensor representation of unique cultural features. Such features, because of their unusual shape, or spectral return, may be prominent landmarks in an area. They could not be properly modeled from DMAAC data alone, but will be modeled from USGS charts, drawings, photographs, etc. In order to keep the data base size within reasonable bounds, and to stay within the limits of an 8000-edge display, many DMAAC cultural features will not, by necessity, be present in the visual data base in high-density areas.

Nevertheless, the use of DMAAC data to generate representative real-world scenes over large areas, in which neither high detail nor extremely accurate placement of features is required, is a very efficient data base generation technique. Thus, LINK will use the DMAAC DDB to automatically generate an initial version of each real-world visual data base for which DMAAC data is available. The resulting data base then will be enhanced

manually, by means of a digitizer, to ensure that significant landmarks are retained and properly portrayed. Cultural features, subtle terrain features, and areas significant for evaluation purposes will be generated through the digitizer and added to the objects automatically generated from DMA data to effect a data base which fulfills the correlation requirements.

The DRLMS data base is produced by much the same process, except that it is modeled on a grid-post system rather than as object lists. The grid intervals used at the finest level of detail are larger than those in the DMAAC DDB; therefore the DRLMS data base differs from the source data, as does the visual data base, but in a different way.

The sensor system does not display the data base at uniform density; rather, it selects certain areas to be displayed in greater detail than others. This is usually done as a function of range to provide greater detail in areas near the ownship, where such detail is most valuable. This process, while performing a very useful function, complicates the correlation of the radar and sensor displays. Radar presentations from the DRLMS are affected by range, but usually in a different fashion than the sensor display. While the level of detail presented by the DRLMS varies with the range selected, the image density is uniform to the sensor horizon. Thus, for example, a distant object that appears complex on the radar display (and in the DMAAC DDB) may look very simple on the sensor display.

As discussed previously, manual intervention is necessary to supplement the transformation program. The purpose of this technique is to increase the detail and placement accuracy of objects and areas for which detail and accurate placement are necessary. In simulated vehicle positions which permit close-up views of cultural objects of terrain areas, the original DMAAC data itself is insufficiently detailed and accurate. In such areas, and in the case of prominent landmarks, manual preparation would be used to enhance the data base, often with the assistance of supplementary data.

3.7 MICROWAVE SENSOR SYSTEMS

The spectral range of microwave radiation runs from wavelengths of about 0.1 centimeter (1,000 micrometers) to 300 centimeters. Atmospheric attenuation of microwave signals occurs, but not in the form of windows as for infrared, but rather with smooth attenuation, decreasing as a function of increasing wavelength. Precipitation attenuation effects occur, also decreasing smoothly as a function of increasing wavelength.

Microwave detectors are composed of a receiving directive antenna system, arranged for sequential scanning of a field-of-view, operated in conjunction with a suitable microwave receiver.

Microwave sensors are of two types: One is a passive system, composed of antenna, receiver, and display to show the microwaves emitted by objects in the FOV, in a manner similar to passive IR systems. The second type is an active, radar, system. Here the FOV is illuminated by locally generated microwave energy, and that portion reflected from objects within this region is amplified and displayed.

3.7.1 Spectral Ranges of Interest

All spectral ranges are of potential interest, but particularly heavy use is made the following bands used for many common types of surveillance/attack radars: X-Band, 3 to 5 centimeters (frequency 5 to 11 gigahertz) and K-Band, 1 to 3 centimeters (frequency 11 to 30 gigahertz). Also interest is strong in M-Band (0.3 to 0.5 centimeters, frequency 60 to 100 gigahertz) for physically small passive sensors.

3.7.2 Sensor Resolutions

Azimuth resolution is a function of the microwave antenna beamwidth in the horizontal direction. A typical value for one centimeter wavelength and 30 centimeters paraboloidal reflector diameter, is 2° beamwidth.

Range resolution for a radar is a function of transmitted pulse length which provides pulsed illumination of the field-of-view. Typical transmitted pulses are of the order of microseconds in length. For a passive sensor, range resolution is a function

of antenna beamwidth in the vertical direction.

3.7.3 Microwave Simulation Algorithm

For the active radar type of microwave sensor system, the basic radar algorithm is shown in Figure 3.7-1. This algorithm is for a pulsed radar with lin-log type of receiver provided with manual gain controls, suitable for a typical surveillance/attack radar system. Simulation of passive sensor systems requires a similar type of algorithm but with values set for a one-way transmission path.

Receiver noise, weather effects, radar beamwidth effects, shadowing effects, low altitude effects, and far shore brightening are not included in this basic equation, but are implemented by subsidiary algorithms in a Digital Radar Landmass Simulator, as described in Section 3.8.

3.8 MICROWAVE IMAGE GENERATOR SUBSYSTEM

This subsystem is composed of a Digital Radar Landmass Simulator (DRLMS). A Radar Effects Generator will be incorporated within the DRLMS. The Data Base Generation/Modification Console in the Mission Management Subsystem will be used, in conjunction with suitable software, called a digital data base transformation program (DDBTP), to transfer DMAAC data to the DRLMS on-line digital data base (DDB).

$$P = K \text{ lin-log} \left\{ \rho \cdot \frac{1}{R^4} \cdot f_1(\Delta R) \right.$$
$$\cdot f_2(K_1 \tan\phi + K_2)\cos\phi$$
$$\cdot \left[K_{IF} + K_{STC}(1-e^{-K_3 R})\right]$$
$$\left. \cdot e^{-K_4 R} \cdot G^2(\phi-\alpha) \cdot G^2_{RD} \right\}$$

Where

P = power at the receiver output

K = proportionality and video gain term

lin-log = receiver transfer characteristic

ρ = reflectance

$f_1(\Delta R)$ = pulse averaging term

R = slant range

K_1 = cosine of the vertical angle between the normal to the element and the normal to the sweep ground track

K_2 = cosine of the horizontal angle between the normal to the element and the sweep ground track

ϕ = depression angle

$f_2(K_1 \tan\phi + K_2)\cos\phi$ = a table which provides for adjustment of aspect effects

K_{IF} = term proportional to IF gain setting

K_{STC} = term proportional to STC gain setting

$(1-e^{-K_3 R})$ = the STC attenuation where K_3 is proportional to STC slope

K_4 = atmospheric attenuation constant

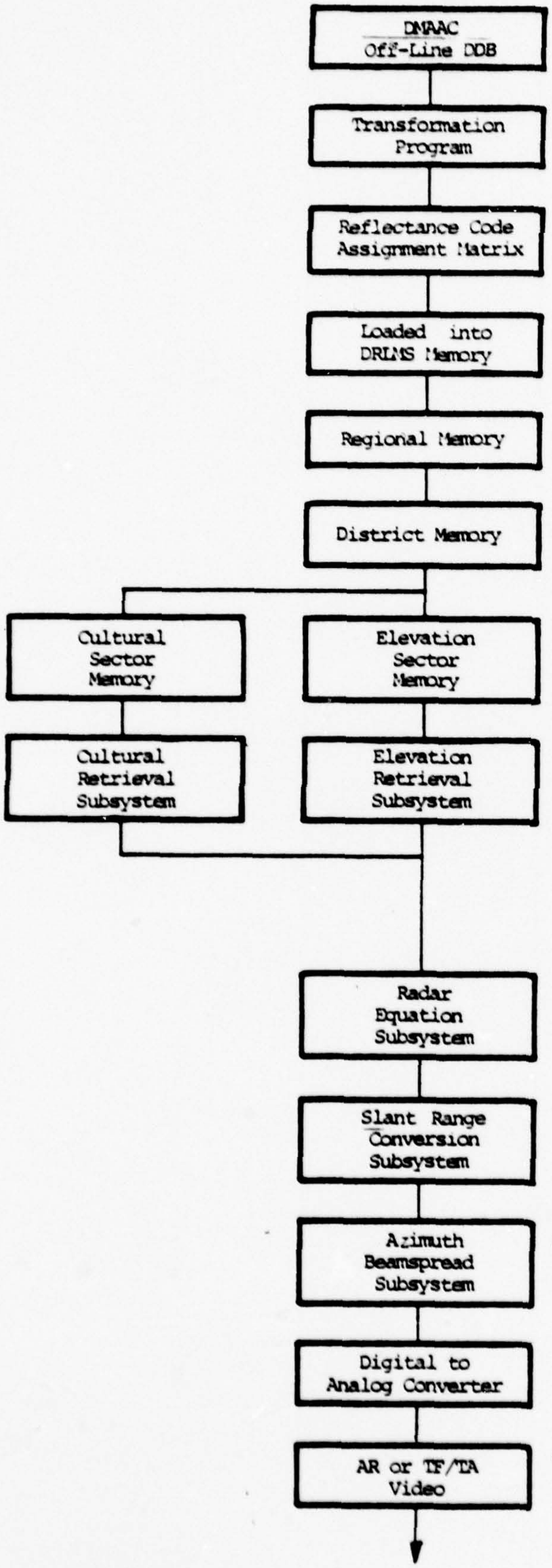
$G^2(\phi-\alpha)$ = antenna gain, where α is antenna tilt

G^2_{RD} = radome gain

Figure 3.7-1 Basic Radar Algorithm

For a DRLMS system to be of interest in the ADSS, the gaming area must be considerably larger than the minimum region which would be useful for FLIR simulation in the DIG device. For this reason the DRLMS system consists of a hierarchy of memories thru which data passes on its way to the display. Each succeeding memory is smaller and faster than the previous one and describes smaller geographical areas. Special purpose computational hardware extracts data for radar sweep lines from the final memory and assigns an intensity to each resolvable element on the display.

Figure 3.8-1 traces the paths of data from DMAAC to the radar display. The regional memory can accommodate a gaming area of up to 1.6×10^6 square nautical miles of radar intelligence by modular expansion of the initial 57,000 square nautical mile area. The district memory, which holds all data within radar range, provides storage for three elevation data bases and three cultural data bases. These multiple resolution data bases service short, medium, and long range settings. Sector memories are updated with the portion of district memory lying closest to the simulated antenna's azimuth. Retrieval subsystems form sweep lines of data from cultural and terrain elevation sector memories which then pass to the radar equation subsystem for computation and integration of discrete display elements.



- Data as prepared by DMAAC
- Produces blocked elevation on Latitude/Longitude grid
- Produces blocked reflectance in List format; reflectivity of features determined by assignment matrix
- Load DDB into regional memory
- Hardware transfer of data in DRLMS
- No alteration or processing of data--only rearranging of data in memory relative to simulated aircraft location
- Lat/Long grid to polar coordinate transformation, parabolic interpolation, earth curvature, Lat/Long list of cultural features to polar coordinates, combination of features into range brackets, computation of horizontal aspect, table look-up of reflectance value and averaging over data elements in one video element.
- The RES computes the video output. Reflectance is only one 8-bit factor in a computation that has a dynamic range of 2^{-128} to 2^{+128} with 16 significant bits. Eight significant bits are extracted from the result and placed in slant range by the SRC.
- The 8-bit value is spread in azimuth in response to the azimuth gain pattern. This is then sent to the D/A converter.
- Processed data displayed on Attack Radar (AR) or Terrain Following/Terrain Avoidance (TF/TA) radar scope (if used).

Figure 3.8-1 Data Path Flow Chart-DMAAC to the Radar Display

Typical requirements are: 30 to 50-foot resolution to 15 nautical miles range; separate computation of horizontal aspect attenuation; means of portraying low-level effects, zero-width objects, and radar reflectors; and vertical terrain accuracy of 50 feet. These display characteristics are produced by first insuring that the data contains all the necessary descriptors and then by providing a real-time geometric computational model that has all the required outputs.

The intensity of each display element is influenced by geometric and electronic factors, but is most strongly influenced by the radar reflectance of the object portrayed. Physical descriptors provided in the off-line DDB by DMAAC are interpreted by the Digital Data Base Transformation Program (DDBTP) into one of sixteen reflectance code assignments. Each code has a modifiable db level assigned in the DRLMS system so that intensity levels can be adjusted in the simulator. Reflectance codes are processed through all the geometric influences appropriate to their placement in range and azimuth, processed through a receiver transfer characteristic model, and finally applied to the display to form an image.

Proper display response requires a full range of simulated effects, from accurate shadows and range attenuation to a faithful receiver model. Among the geometric effects, the requirement for a rigorous solution to the effect of horizontal aspect and the orientation of features has been the most influential on the form of the on-line data and the method of assembling sweep lines of data. A fully three-dimensional vector algebra

solution for the angle between incident radar energy and the normal to the terrain or cultural surface is implemented in the system to produce a complete aspect effect.

Data which can contribute to the display is moved from regional memory to district memory in response to aircraft movement, and from district memory to sector memory in response to radar antenna movement. The data is divided into quadrangular blocks for convenience in assembling the district and sector geographic coverage. These data movements are controlled by the computer complex. Once data is in sector memory, special purpose hardware extracts sweep lines of data - one sweep line of natural terrain elevation and a matching sweep line of cultural data to be superimposed on the terrain. The data passes through the radar equation subsystem where geometric and receiver effects are calculated, then through the azimuth beamspread subsystem for azimuth integration and convolution with the antenna azimuth gain pattern, then is converted to an analog signal which modulates the video input line of the radar indicator.

The process of data retrieval consists of collecting data elements from sector memory along a sweep line and range ordering them for further processing.

The retrieval process for terrain elevation data is quite different from that for cultural data because the data structure is different. Elevation data is arranged in a grid structure in latitude and longitude. There are three subsets of elevation

data having 3-second, 6-second, and 12-second grid pitch, respectively. The three data subsets are present everywhere in the gaming area and are used to support short, medium, and long radar range settings, respectively. To assemble a sweep line of elevation data, which is essentially a profile of the terrain along a sweep line, sector memory data along the sweep line's ground track is read directly from sector memory. Sweep line elements that fall between the data grid points are interpolated from a 4 x 4 matrix of elevation grid values. This process proceeds in range order from nadir to end-of-range for every sweep line. Since the sweep line's ground track lies on a great circle, and the data is on a geocentric grid, the sweep heading must be corrected at intervals to stay on track. This correction is accomplished automatically as the calculations proceed. At the output of the elevation retrieval process, earth curvature correction is added to the terrain profile. Finally, comparisons of sequential sweeps yield solutions to the horizontal and vertical components of the normal vector to the terrain, used later to determine the net aspect angle.

The cultural retrieval process does not proceed in range order because the data is directly addressable only by quadrangular blocks, not by data elements. Within each block cultural data is listed in terms of aerial perimeter points for each feature along with reflectance and cultural elevation descriptors. The processor must analyze all the data in each candidate block to determine placement of data elements in range and azimuth, and

then must reject elements which do not intersect the sweep line of interest. Range-azimuth solutions for points stored in latitude-longitude coordinates require the solution of a spherical triangle having as vertices one pole, the aircraft, and the data point. The equations for this solution are implemented in a high-speed processor which produces a solution every 200 nanoseconds. Subsequent processors sort the data by range and azimuth and fill in the voids between data intercepts so that the final range ordered cultural data sweep lines contain a reflectance, cultural elevation, and aspect description for each sweep line element. As many as five data samples may be averaged to form one sweep line element.

The radar equation subsystem (RES) calculates the geometric and electronic effects which act on the data provided by the retrieval subsystems. Each data element is assigned a slant range (R) and a depression angle (Ø) by trigonometric computation from the data element ground range and elevation. Then the equation of Figure 3.7-1 is executed to compute display intensity for each data element. Receiver noise, weather effects, shadowing, low altitude effects, and far shore brightening are not shown in the equation but are implemented by subsidiary algorithms.

In the final processing of four-bit reflectance codes in the culture retrieval system, multiple elements are averaged together to form an eight-bit number for processing by the radar equation subsystem. The eight-bit number is $\log_2$ with a five-bit characteristic and three-bit mantissa. The range of the

number represented by the eight bits is 2^0 and 2^{-32}, which yields a dynamic reflectance range of 0 to -96 db.

Any of the 16 reflectance codes, represented by a four-bit number in the data base can be assigned one of 256 values between 2^0 and 2^{-32} (to the nearest 3/8 db) by means of tables located in the cultural retrieval subsystem. These same tables perform the averaging function so any number in the specified range can emerge from the averaging process.

The logarithmic reflectance thus produced enters the radar equation subsystem where it is summed with other logarithmic geometric attenuation factors such as aspect angle, range attenuation, etc. *When a return is attenuated* below Minimum Discernible Signal (MDS), it does not drive the display, and when a return saturates the receiver, it drives the display at full brightness.

Digital computation with logarithmic algorithms yields exceptional dynamic range. The numbers carried in the RES could accommodate a radar set having a transmitter peak power of 10^{35} watts and a receiver MDS of -384 dbm.

To simulate the linear-log transfer characteristic of the receiver, the K_{IF} term shown in the radar equation is manipulated in response to IF gain setting. For maximum IF gain, K_{IF} is set to +115 db, effectively shifting the gain transfer function.

The video gain control also shifts the transfer function. At a minimum video gain setting a signal 80 db over MDS saturates

the display. The gain term K is manipulated as a function of video gain to produce this effect.

The radar equation subsystem computes shadows by successive comparisons of depression angles (ø) as the sweep proceeds from nadir to end of range.

Distinctive features are encloded in the data base by storing their perimeter points, reflectances, and elevations above the terrain. Any effects attributable to the nature of the feature itself or its surroundings (as in the case of farshore brightening), such as low-level visibility or directionality are produced in addition to the normal effects of beamspreading, range attenuation, etc. by retaining the feature's identity up to the point in the processing where the effect is produced. For example, radar reflectors are retained until the radar equation subsystem can make a visibility determination from aircraft altitude.

The RES calculates ground return intensity in increasing increments of ground range. Operational radar systems, however, and the operational aircraft indicators selected for simulator use, process and display returns in linearly increasing increments of time, or slant range. The SRC subsystem performs the time-base conversion of display intensity levels from ground range to slant range geometry by placing the intensity level data in a high-speed memory ordered in slant range.

For flat terrain and small depression angles, ground range (GR) and a slant range (R) are approximately equal; also, small changes (Δ) in GR and R are approximately equal. For uneven terrain and for large depression angles, this relationship no longer holds. Instead, there can be a virtually random relationship between GR and R, and between ΔGR and ΔR; even the sign of ΔR can change.

In the actual radar system, the cases where $\Delta R=0$ or changes sign cause very bright displays, since each slant range increment receives power from multiple ground range locations (equivalent to a larger area than normally illuminated). To simulate this power buildup, the SRC is designed to accumulate power from successive ground range data points in a high-speed memory whose addresses represent equal slant range increments. A number of data points are permitted to contribute power to a given memory word. When the memory is read out sequentially, the result is a stream of accurately simulated power returns from equal increments of slant range.

Another effect which is simulated in the SRC is pulsewidth stretching. This effect occurs because the radar pulse is of discrete length; therefore the return from a single target is spread over a period of time greater than the width of the target, namely one-half the pulsewidth. In the DRLMS, both slant range conversion and pulse stretching are performed at the same time by adding c/2 (pulse stretching width) to the slant range location.

The azimuth beamspread subsystem (ABS) simulates the azimuthal spreading effect of the antenna horizontal gain pattern on the radar return. The putput of the RES, ordered in slant range by SRC, is the input to the ABS. The ABS calculates for display a weighted average of all sweep lines within the beam limits of the antenna. The number of sweeps stored is determined by the following relationships:

$$\frac{\text{PRF}}{\text{Scan Rate}} \text{ X Beamwidth} = \text{Number of Sweep Lines}$$

In one embodiment of ABS each sweep line memory is a high-speed shift register, each long enough to contain the data for a complete sweep. Since the antenna is scanning it is necessary to move a given sweep line through the beamwidth. At each main trigger pulse, data shifts completely through each register and is reloaded into one of the adjacent registers. The side chosen depends upon the direction of antenna scan. The data in the register at one edge of the beamwidth is dropped, making room for one new sweep at the other edge of the pattern. The output of the registers are presented in parallel to a weighted summation network. The weight given to each shift register output is determined by the antenna azimuth gain pattern. This pattern is symmetrical around the center of the beam.

The ABS output passes to the analog subsystem for D/A conversion and any required analog video modification effects, and then to display CRT.

3.9 RADAR/SENSOR CORRELATION IMPROVEMENT

Section 2.2 provided a description of the current methods of generating DIG and DRLMS data bases, and indicated the reasons for existence of residual uncorrelated errors between the displays of the two devices. For applications requiring very low values of residual uncorrelated error, it is important that targets, threats, radio navigation stations, and major terrain and cultural features be correlated in position on the display of the DIG device (for FLIR and LLLTV), the display of the DRLMS and suitable displays and indicators for radio navigation, and electronic warfare, if used.

In the ADSS these devices are separate computer complexes, each with a data base of appropriate form for a selected gaming area. These data bases differ in format and detail for each device, such as DIG, DRLMS, RN, EW, etc. As an example, Figure 3.9-1 shows the relative scene detail required as a function of range for DIG and DRLMS data bases.

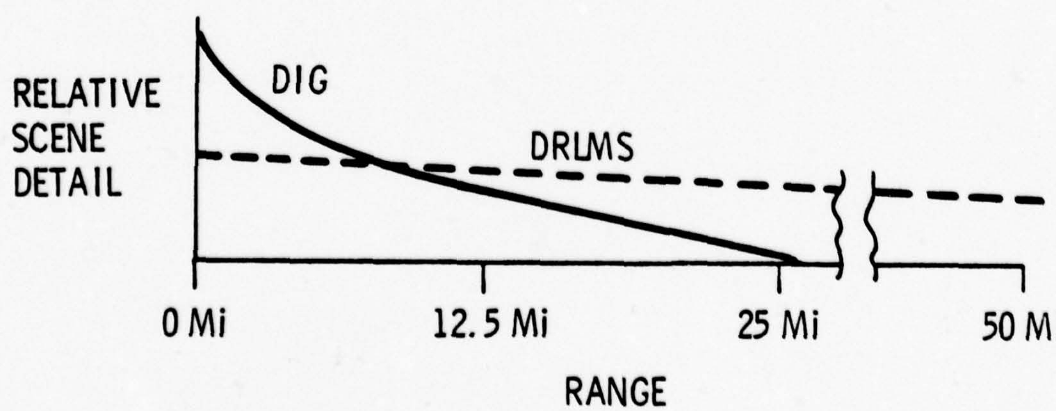


Figure 3.9-1 RELATIVE SCENE DETAIL

A proposed method of insuring that this correlation among data bases exists is shown in Figure 3.9.2. As shown, the input data from DMAAC is automatically transformed, while input data from other sources is semi-automatically transformed to produce a Master On-Line Digital Data Base. This base contains all required levels of detail of all scene elements which may exist within the gaming area, and provides position, elevation, reflectivity, emissivity and any other required data. Then individual transformation/compression programs operate on this master DDB to form the individual DRLMS DDB, DIG DDB, etc. Correlation of scene elements derived from the one common master DDB then will be within the computational and display capabilities of each of the subsystems.

4. CONCLUSION

A description of the characteristics of potential ADSS devices has been provided. There appears to be a good probability that such a device can be successfully designed and constructed in a manner compatible with the descriptions of Sections 2 and 3, to permit successful implementation of the sensor system test and evaluation objectives stated in Section 1. This conclusion is supported by the photographic exhibits furnished under this contract, whose titles are listed in Appendix I. These exhibits show tactical scenes taken directly from the viewing screen of a LINK inhouse digital sensor simulation system. Appendix II then provides a suggested version of an "Advanced Digital System Simulator Specification" document.

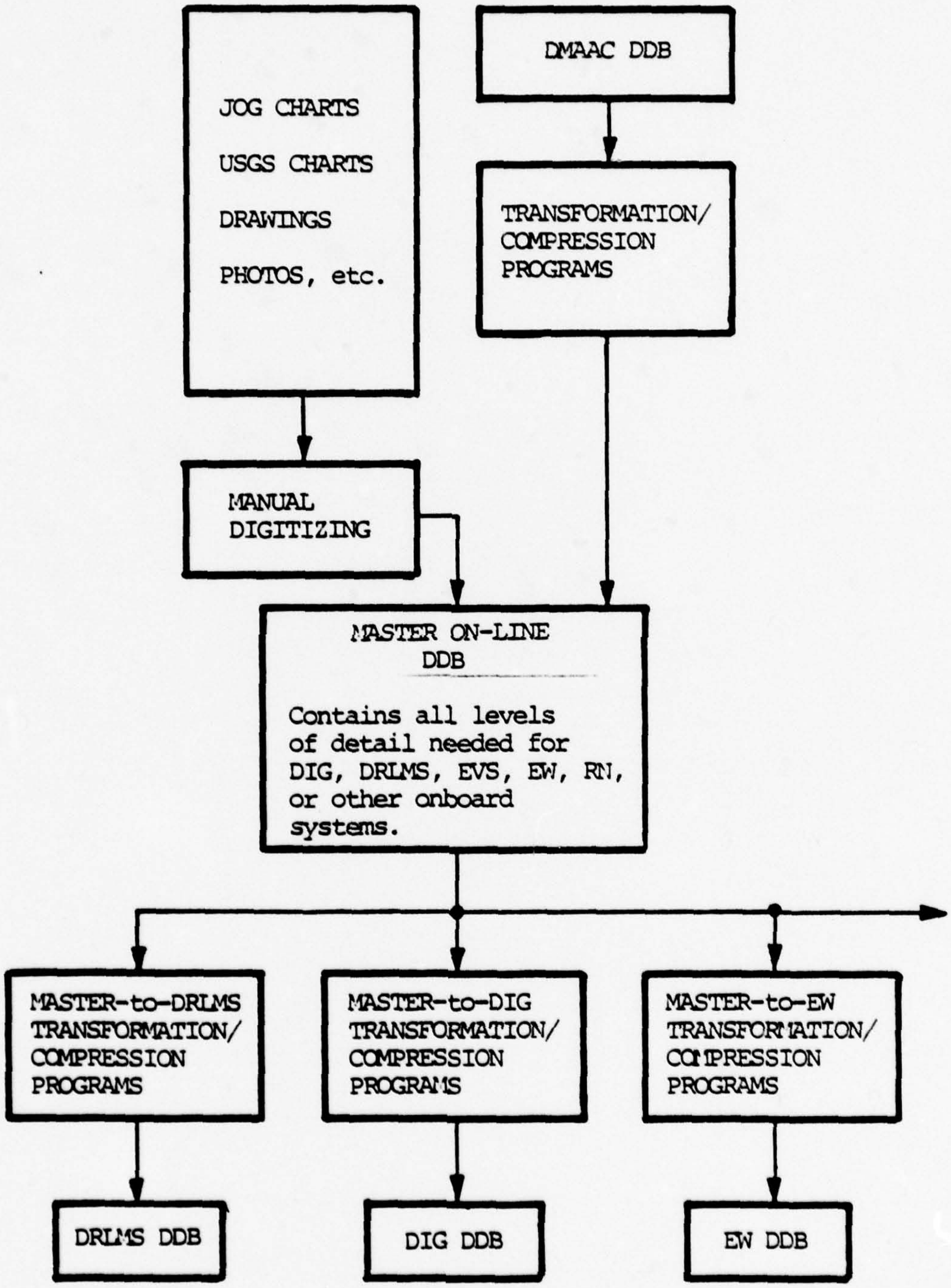


FIGURE 3.9-2 DATA CORRELATION FLOW DIAGRAM

APPENDIX I

ADVANCED DIGITAL SYSTEM SIMULATOR
DIGITAL SCENE PHOTOGRAPHY

Prepared For

U. S. ARMY ELECTRONICS, RESEARCH & DEVELOPMENT COMMAND
NIGHT VISION AND ELECTRO-OPTICS LABORATORY
FORT BELVOIR, VIRGINIA

CONTRACT DAAK70 77 C 0175

Prepared By

SINGER/LINK DIVISION
1077 E. ARQUES AVENUE, SUNNYVALE, CALIFORNIA 94086

JULY 1978

APPENDIX I

As part of the ADSS study a number of digital scenes were constructed and photographed. These illustrated some of the more difficult aspects of the program and were taken directly from the LINK R&D Digital Image System. They were supplied to Nick Diakides of the NVL.

1. 16 mm movie showing typical terrain and a West European village along with moving tanks, moving jeeps and a moving helicopter.

2. Twelve 4 x 5 color pictures showing terrain, trees of various kinds, buildings, canals, bridges, tunnels, high detail tanks and other military vehicles.

APPENDIX II

ADVANCED DIGITAL SYSTEM SIMULATOR SPECIFICATION

Prepared For

U. S. ARMY ELECTRONICS, RESEARCH & DEVELOPMENT COMMAND
NIGHT VISION AND ELECTRO-OPTICS LABORATORY
FORT BELVOIR, VIRGINIA

CONTRACT DAAK70 77 C 0175

Prepared By

SINGER/LINK DIVISION
1077 E. ARQUES AVENUE, SUNNYVALE, CALIFORNIA 94086

JULY 1978

ADVANCED DIGITAL SYSTEM SIMULATOR SPECIFICATION

1.0 INTRODUCTION

The Advanced Digital System Simulator (ADSS) shall be a device which will permit operator evaluation of simulated electro-optical viewing systems (EVS) under conditions similar to field usage. The ADSS shall simulate the operation of the following two types of systems:

A) Helicopter-borne sensor systems for threat identification or neutralization.
B) Fixed-wing Remotely Piloted Vehicle-borne sensor systems for threat identification or neutralization.

The ADSS device shall provide effective simulation of specific members of the above classes of systems, together with simulation of a selected battle area composed of terrain features, target characteristics, target actions (both threat and evasive), ownship responses to flight controls and to offensive/defensive controls (laser designator, missile guidance), together with daily and seasonal weather effects.

The utilization of the ADSS device to test proposed new members of Type A and Type B systems shall be through use of Mission Scenarios, which are simulations of typical sequences of events in real-world battle areas. Each Mission Scenario needs to provide operator-selectable lists of available parameters as follows:

1) Sensor: Parameters such as resolution, bandwidth, camera effects, radio link effects (where applicable), display effects, etc.
2) Environment: Parameters such as terrain and cultural data base selection, target types, numbers of targets, and actions of targets.

When these selections have been made, from a preprogrammed list of available alternates, the particular mission scenario to be used in a test sequence will have been determined. New additions to the lists of available parameters can be added as desired through a Data Base/Scenario Modification Console.

2.0 SCOPE

The contractor shall design, develop, fabricate, integrate, and test an Advanced Digital System Simulator which includes a visual system and control computation system.

The ADSS includes the visual display, evaluator station, computer image generation system, control computer(s), operating console, and peripheral equipment. The contractor shall provide all computational equipment and software necessary to drive the evaluator station and visual system in a manner suitable for evaluating the capability of the sensor system being simulated. After delivery and acceptance of the ADSS device by the government, the contractor shall provide maintenance and spare parts for a six (6) month period. This project is to be accomplished in a single contract with suitable milestones established to track and to evaluate the contractor's progress. The contractual effort will cover a period of 24 months.

2.1 Design, develop, and fabricate a mission management/subsystem which shall consist of a mission management computer complex, mission operator/ evaluator console, data base/scenario modification console, flight/navigation instruments and controls, and test subject station.

2.2 Design, develop, and fabricate an image generator subsystem, which shall consist of a digital image generator (DIG), a sensor effects generator (SEG), and a CRT display monitor, arranged to simulate the outputs of various EVS sensors, such as infrared, low-light-level TV, and microwave, in both passive (natural illumination) and active (self-generated illumination) modes.

2.3 Design, develop, and fabricate an offensive/defensive/subsystem, which shall consist of offensive/defensive instruments and controls, an offensive/defensive display generator, and suitable display devices.

2.4 Provide a data base describing terrain and ground target features for a prescribed area. The contractor shall also provide a data base management system (DBMS) which shall include the data base/scenario modification console specified in section 2.1.

3.0 GENERAL ADSS TECHNICAL REQUIREMENTS

The contractor shall design, develop, fabricate, integrate, prepare for in-plant testing, test, disassemble, ship, reassemble and install the ADSS device, including the mission management, image generator, and offensive/defensive subsystems, in accordance with the following.

3.1 The mission management subsystem is composed of a computer complex, the mission management computer and associated peripherals, configured in accordance with the requirements of section 4.0, a mission operator/evaluator console, a data base/scenario modification console, and instruments and controls for operation of the ADSS by the test subject.

3.1.1 The operator/evaluator console shall be composed of a CRT page display and a function keyboard interfaced to the mission management computer. The CRT page display shall consist of tabular displays of alphanumeric characters and a small number of special symbols. A generator for these characters and symbols, a buffer memory, and the necessary raster and video generation equipment needed to refresh the display shall be included.

3.1.2 The data base/scenario modification console shall consist of a CRT display with graphic and alphanumeric capability and function keyboard interfaced to the mission management computer. It shall also include an X-Y digitizing table, a hard-copy generator for the CRT screen content, and a line printer. The purpose of this equipment is the generation or modification of mission scenarios and digital data bases which describe the prescribed gaming area terrain, cultural objects, and target objects, as described in a three-dimensional coordinate system.

3.1.3 The test subject instruments and controls shall simulate the equivalent devices for the Type A or B sensor systems.

3.2 The image generator subsystem is composed of a digital image generator (DIG) a sensor effects generator, (SEG) and a CRT display monitor.

3.2.1 The digital image generator (DIG) shall use a digital data base which describes a prescribed gaming area of points, lines, polygons, and other primitive forms, to generate an image of the portion of the gaming area visible from any arbitrary electro-optical sensor viewpoint.

3.2.1.1 The DIG shall be composed of a commercially available digital computer and appropriate peripheral devices, configured in accordance with the requirements of Section 4.0, together with special-purpose digital computation equipment, and video generation equipment.

3.2.2 Image Requirements. This section describes the image to be generated by the DIG.

3.2.2.1 Scene Construction. The DIG system shall compute the visual scene by transforming the three-dimensional model, stored in memory, into a two-dimensional display scene oriented for the selected viewing point. This transformation shall include target positions and attitudes within the modeled visual data base. The result of the computation shall be a video signal which is tied directly to the sensor Effects Generator.

3.2.2.2 Edge Definition. The DIG visual scenes, will consist of patterns arranged to appear as recognizable scenes. These patterns shall consist of planes of different brightness. Individual shades will be delimited by edges. The number of edges which can be processed and displayed is a measure of the system performance. For the purpose of this specification, the term "scene edge" is a line which separates two shades on a plane or is a line which defines the intersection of two or more planes.

3.2.2.3 Edge Retrieval. The data base memory shall contain the necessary data for describing the simulated visual environment. The DIG system shall, from data base memory, retrieve and process only the data in the immediate locality of the sensor vehicle position, which is provisionally defined as a minimum of 3 km from the vehicle position. Thus, the number of stored edges can be many times the number of edges that are processed and displayed as the vehicle moves within the visual data base. The deletion or addition of new data being processed shall not be discernible at the display eyepoints.

3.2.2.3.1 Processed and Displayed Edges. The processing of visual content and detail shall vary with range, magnification and sensor. The DIG system shall include provisions so that edges which enclose an area smaller than the area of one display resolution element shall not be processed, except for those areas whose influence is important to the scene. The DIG system shall be capable of processing and displaying,

after overhead considerations, a minimum of 8000 potentially visible scene edges every television frame time. Potentially visible edges include those edges on solid objects which are in the field-of-view on the sides of the objects facing the view point. A scene edge is potentially visible if it is occulted by an object but would be visible only if that object were moved.

3.2.2.3.2 Nonvisible Edge Elimination. To increase the number of visible scene edges, the DIG system shall eliminate the nonvisible edges during the initial stages of processing. The nonvisible edges are those edges on solid objects which are totally outside the field of view, and those edges on solid objects which cannot be seen because they are on the opposite side from the eyepoint.

3.2.2.4 Level of Detail. Each object or group of objects in the visual data base shall be modeled in different levels of detail. These levels of detail shall range from maximum image complexity to elimination from processing. The DIG system shall use these levels of detail to produce the maximum scene complexity to the eyepoint during the entire simulated mission. The minimum criteria for changing level of detail of an object or group of objects shall be as follows:

a) The range from the eyepoint to the object or group of objects.
b) Maintaining the maximum number, or near maximum number, of displayed edges.

The insertion of deletion of objects or edges shall be gradual to eliminate the "popping" in or out of visual information.

3.2.2.5 Scene Characteristics. The visual scenes that are generated by the DIG system shall be in proper perspective on the display. This includes:

a) The position and perspective of the horizon and of all objects.
b) Occultation of distant images and moving objects by nearer objects.
c) Contrast reduction as a function of range and visibility.

3.2.2.5.1 Objects. Objects shall be solid face polyhedra, affixed to the surface representing terrain features, cultural detail, enemy emplacements, etc.

3.2.2.5.2 Moving Objects. Moving objects shall be capable of independent movement on or above the surface, representing both aircraft and ground vehicles. The moving objects shall have complete freedom of positional and angular movement within the visual data base, but shall be limited to preselected and preprogrammed paths.

3.2.2.5.3 Curved Surface Shading. The capability of continuous shading of surfaces shall be incorporated in the DIG system. This capability shall remove the visibility of the edges of the polygons which make up the surface and give this surface the appearance of being curved. The continuous shading of surfaces shall, as a minimum, use a fixed sun position for the generation of the surface shading. The effects of the fixed sun position on moving object surfaces shall be computed in real time. This enhancement is not to be applied to all objects displayed, but rather only to those objects that are identified as curved in the real time visual data base.

3.2.2.5.4 Surface Texture. Texture shall be added to the surface for providing surface composition, velocity, and change-of-velocity cues to the eyepoint. This texture shall remain stationary with respect to the surface and shall maintain the proper perspective, occulting, and contrast reduction of the surface. The texture shall not be repetitive across the total surface but may be repetitive in local areas representing fields, one type of terrain or one class of surface area.

3.2.2.5.5 Sky. The sky shall be of a uniform but selectable brightness, to be displayed where no surface plane or object is being displayed. This shall be changeable by computer software. The sky shall fade according to horizon haze, visibility, and ambient light level selected.

3.2.2.5.6 Point Lights. Point lights shall vary in brightness as a function of range and visibility, and shall be a minimum size of one resolution element or less if, required to produce a more acceptable image. These lights shall be used to represent cultural detail lighting.

3.2.2.5.7 Line Features. Special objects known as line objects shall be provided by the DIG. These objects shall be such that the width of the object is a constant one or two picture elements wide regardless of the range to the object.

3.2.2.6 Suppression of Distracting Visual Effects. The DIG shall suppress distracting visual effects that might otherwise occur during the computation and processing of the image. These include:

a) Scintillation of small faces
b) Quantization due to computation of picture elements
c) Abrupt addition or deletion of scene detail

3.2.2.7 Transport Delay. The elapased time from a vehicle station control input to response of the visual scene shall be not less than the control system response time, nor more than the control system response time plus 100 milliseconds.

3.2.2.8 Display Update and Refresh Rate. The DIG will process new position data each 1/30 of a second. The scene shall be refreshed at a rate of 60 fields/second.

3.2.2.9 Intensity Control. At least 256 logarithmic levels of intensity shall be available where the dynamic range shall be 256:1. The intensity of each surface face and light point shall be computed each frame as a function of the base intensity level, simulated distance from the vehicle, sun position and visibility setting.

3.2.2.10 Number of Edges. The DIG will generate 8000 potentially visible edges or 8000 potentially visible lights or any combination of edges and lights totaling 8000 edges. All lights shall count as a single edge per light. The number of potentially visible edges is computed by determining the maximum number of potentially visible edges the DIG can handle and then subtracting all overhead processing loads.

3.2.2.11 Intersections. The scanline computer shall permit a maximum of 512 intersections per scanline. The 512 limit shall be maintainable over each and every scanline.

3.2.2.12 Area of Coverage. The DIG shall be capable of processing a minimum of 10 X 10 nm.

3.2.2.13 Level of Detail. The DIG shall process at least four levels of detail. These levels will include the "elimination from processing" level.

3.2.2.14 Moving Objects. The DIG shall be capable of generating a sufficient number of moving objects to simulate up to ten (10) independently moving vehicles for simultaneous display. The moving objects shall include but not be limited to enemy tanks, trucks and aircraft.

3.2.2.15 Scene Resolution. The DIG shall be capable of generating input data for a display monitor in any one of three formats. These are square with the number of picture elements per scanline equal to the number of scanlines per frame; or with the number of picture elements per scanline equal to 4/3 the number of scanlines; or with the number of picture elements per scanline equal to 3/4 the number of scanlines. The desired format shall be switch selectable. Also, four values of scanline number shall be switch selectable, with other discrete values of scanlines per frame available by changing coding of PROM units and placing them in appropriate sockets, which then permit this value to be switch-selected. Typical possible values of scanlines per frame are: 256, 512, 525, 630, ... 1,000 maximum.

3.2.3 Special Effects

3.2.3.1 Illumination Levels. The DIG data base must provide the following values for each object face:

For Passive IR/sensor simulation: Emmissivity for day and night illumination, also emissivity for winter, summer, dry or rain-covered. Emissivity values are to be provided for 3-5 micron and 8-11 micron windows.

For Passive millimeter-wave sensor simulation: Similar to IR, with values provided for 94 gigahertz (GHz) window.

For Active IR simulation: IR reflectivity values when illuminated by IR scanner or search-light, for 3-5 and 8-11 micron windows.

For Active millimeter-wave simulation: Same, for 94 GHzwindow.

3.2.3.2 Variable Weather. The DIG will provide varying visibility ranges. Selection of a visibility range shall be performed at the instructor/operator station (IOS) and shall range from 0-10,000 feet in 50 foot intervals.

Dynamic rain is required and will be controllable at the IOS.

Snow storms are not required but snow scenes (snow on the ground) are required and will be selected at the IOS.

3.2.4 Data Base Size

The maneuvering area for the sensor shall be 10 X 10 nm. The actual data base shall be enlarged with a low detail scene such that the edge of the data base cannot be seen from any point within the maneuvering area.

3.2.5 Edge Density

The goal of the visual system shall be to maintain the maximum edge density in the scene at all times. As a minimum

requirement there shall be no point in the data base at which less than 4000 edges are visible (except at border).

3.2.6 Seasonal Effects

Data bases must be prepared for both winter and summer scenes.

3.2.7 Moving Objects

The data base shall contain moving objects which can be selected at run time and moved throughout the data base on preprogrammed paths. Path selection and velocity for each moving object shall be controllable by the operator, up to a maximum of ten (10).

3.2.8 Movable Objects

The data base shall contain a library of movable objects from which the operator can select and place a maximum of ten objects prior to each mission. These movable objects will consist of such items as parked vehicles, personnel, and fortified positions.

3.2.9 Weapon Effects

Gun flash and missile plume effects shall be simulated. The size and intensity of the flash shall be a function of the line of sight and the position of the gun relative to the viewpoint of the sensor platform.

Tracers shall be simulated. Trajectories and hit points shall be computed. This load shall be shared between the DIG controller and the Mission Management Computer.

The DIG shall simulate the weapon impact by displaying appropriate flash and smoke effects at the point of impact. The shape of the effect shall be a function of the type of round fired and the type of object which was hit. Any smoke effects shall be properly formed and shall move and dissipate as a function of the wind velocity and direction.

The impact of the laser designator beam shall be shown when it is within the field of view and the selected atmosphere window response of the sensor.

3.2.10 Sensor Effects Generator

The type of sensor effects generator (SEG) required is an electro-optical viewing system (EVS) Video Effects generator.

One of the most critical aspects of EVS simulation is that of providing realistic simulation of the nonideal characteristics of the EVS sensors. Several characteristics which commonly affect the output of electro-optical image sensors are:

1. Signal-to-noise ratio
2. Resolution and/or focus
3. Threshold and limiting
4. Nonlinearity
5. Blooming
6. Overload
7. Automatic gain control or iris control
8. Slow image decay
9. Sensor element mismatch

The SEG system is to employ three primary video paths. First is the forward path which is used for inserting those effects which may be added by simple electronic signal processing means.

These effects include signal-to-noise ratio, threshold, limiting, nonlinearity, gain control, and sensor element mismatch. The second path provides a forward video path through a display and camera pickup. This path provides for inclusion of the effects of resolution and/or focus, blooming, and overload. The direct electronic forward path bypasses the display/camera chain in order to provide nondegraded video at the output which is then mixed with the camera output. The third path provides video feedback to the display from the camera. This path provides for simulation of slow image decay effects.

3.3 OFFENSIVE/DEFENSIVE SUBSYSTEM

This subsystem shall be relatively small and uncomplicated since its computational load shall be supported by the DIG control computer. The details of controls and displays shall be specified after the following information is determined:

Laser Designator:	If on ownship, exact type. If on ground or other vehicle, power, beamwidth, wavelength, and any coding used.
Missile Guidance:	If on ownship, exact type.

4.0 DIGITAL COMPUTATION SYSTEM

The digital computation system shall consist of one or more commercially available general purpose digital computer configurations with the capabilities, software, and peripheral equipment specified herein. Consideration in selection of the computation system shall include reliability, flexibility, accuracy, maintainability, ease of programming and debug, quality of support software such as compilers, assemblers, loaders, input/output handlers and operating system, and compatibility with the DIG computation system. The digital computation system shall accept control inputs, perform a real-time solution of the ADSS system mathematical model, and provide outputs necessary to accurately represent the

static and dynamic behavior of the simulated real-world systems within the tolerances and performance criteria specified herein. The computational system equipment and computer programs shall have the flexibility and capacity to facilitate programming modifications and additions.

4.1 Computer System Equipment

The selected computation system equipment shall be capable of performing all real- and nonreal-time processing tasks required and shall, as a minimum, possess the specific features described below.

4.1.1 Multiprocessor/Multicomputer Configuration. It is anticipated that a multiprocessor/multicomputer configuration shall meet the ADSS requirements. The following shall apply to each computer/processor (control computer and general purpose portion of DIG) in the system. Multiprocessor/multicomputer configuration shall be designed such that all computers are controlled and time-synchronized from a single computer program supervisor/executive. This supervisor/executive shall direct the problem flow and establish priority controls. Each computer shall be capable of communicating with 1) all peripheral equipment, 2) all operating console systems for which it computes information or controls information flow and 3) all interfaces for which it computes or controls information. Standard computer vendor equipment shall not be modified. Each computer shall be capable of transferring or receiving data/information directly to or from other computers. A commercially available executive control program shall be used to control the multiprocessor system by directing problem flow and establishing priority controls. The configuration shall provide for efficient interface with the DIG computation system.

4.1.2 Computer Main Frame Characteristics. The computer equipment shall provide the memory capacity, word size, speed, and input/output capability compatible with the efficient realization of the total ADSS system. The computer equipment shall include the following features (these requirements apply to each computer main frame in the computation system).

a. Power fail safe and automatic restart.
b. Interval timer to measure elapsed CPU time or real time with at least 500 nanoseconds resolution.
c. Priority interrupts for all peripherals and input/output channels

d. Hardware bootstraps(s) for prime object loading peripheral magnetic media device(s), including magnetic disk and magnetic tape.

e. Word length to be 32-bit or greater, and sufficient to:
   (i) Handle a floating-point word, including sign, fraction, and exponent, in a single memory location
   (ii) Preclude the necessity for using double-precision arithmetic in the simulation program
   (iii) Provide optimal compatibility with the DIG computation system.

f. Memory to provide:
   (i) speeds faster than 800 nanoseconds
   (ii) at least four-way interleaving
   (iii) memory mapping

g. Hardware floating-point arithmetic

h. Instruction timing for single-precision floating-point under
   (i) 2 microseconds for addition and subtraction
   (ii) 4 microseconds for multiplication
   (iii) 4.5 microseconds for division

i. Instruction look ahead capability

j. Writable control storage

4.1.3 Input/Output Capability. All I/O processors shall be external to the CPU and shall have sufficient speed and channel capacity to enable input/output operations to proceed in real-time without disrupting or degrading real-time simulation/ operation. To satisfy this requirement, each computer input/ output system shall include, but not be limited to, the following capabilities and provisions:

a. Each I/O processor shall have its own data path to memory (port), independent of the CPU or other I/O processors. This mode of input/output shall allow external devices to transfer blocks of data to or from the computational system memory without CPU intervention for each transfer. Each I/O processor shall operate asynchronously.

b. Capability to input or output, while in real-time simulation operation, to all units of peripheral equipment including magnetic disk and tape.

c. Means by which the computer can be externally interrupted.

d. An error checking feature to check input/output transfers.

e. Capability to directly communicate, during real-time operations, with those interfaces for which the computer processes and controls information.

4.1.4 Interrupts. The computer shall provide the capability to service interrupts in a priority structure. The hardware of each computer shall provide a means of distinguishing between simultaneous interrupts and provide the computer programs a means of entering service routines for interrupt processing. At least 50 levels of interrupts shall be available, of which 32 levels are external.

4.1.5 Computer Control Panel. A computer panel shall be provided for each processor included in the computation system configuration. Each panel may be located on a separate console, or may be incorporated into a centralized panel location on a computer rack. Capability for manual insertion of instructions and data shall be provided at both the control panel or panels, and the input typewriter.

4.1.5.1 Register Information Display and Insertion. Display indicators and associated controls shall be provided to allow selection and visual examination of the contents of any program-accessible register. Hardware switches and associated controls shall be provided to permit insertion of information into any program-accessible register.

4.1.5.2 Halting Provisions. Means shall be provided to halt the computer at any preselected program step. A means shall also be provided to optionally halt or not halt upon the occurrence of a parity error.

4.1.5.3 Parity Error Indicators. Indicator lamps shall be provided to provide a visual cue of the occurrence of a parity error.

4.1.5.4 Single-Step Provisions. Single instruction stepping of the program through the processor shall be provided as an aid in locating computer malfunctions and debugging programs.

4.1.6 Interface Equipment. The interface equipment shall be designed to permit the activation of the evaluator station and sampling of controls with sufficient accuracy and speed to minimize I/O signal error and eliminate discernible discrete stepping of indicators and other appropriate outputs. Standards for form and format of I/O quantities shall be established for the computer side and evaluator station side of the interface. The interface system shall include analog-to-digital input conversion equipment, digital-to-analog output conversion equipment, discrete inputs, discrete outputs, digital inputs, digital outputs, digital-to-synchro outputs, etc., and necessary control equipment. The interface equipment shall interconnect directly with each computer with which it communicates directly, i.e., the computer in which information originates or terminates.

4.1.6.1 Converters. Digital-to-synchro converters shall be used for converting digital information to angular shaft positions. Analog servo equipment shall not be used in the interface equipment.

4.1.6.2 Electronic Relays. Electronic relays or switches shall be used for discrete control functions. Mechanical relays shall not be used in the interface system.

4.1.6.3 Converter Resolution. The input converters shall be capable of resolutions of one-half the accuracy tolerance specified for the particular control. The resolution of the output converters shall satisfy, as a minimum, the following equation:

$$n = 6 + 3.3 \log_{10} S$$

where:
n = converter word length in bits
S = instrument scale length in inches

4.1.7 Peripheral and auxiliary equipment shall be provided to support the computational system in meeting the requirements of the ADSS.

4.1.7.1 Card Reader. Minimum read speed shall be 600 cards/ minute.

4.1.7.2 Typewriter Keyboard and Print-Out Device. This device shall be designed for heavy continuous use as the primary communication device for computer control.

4.1.7.3 Magnetic Disk System. A magnetic disk system shall be the primary system for storing and for loading computer memory with all ADSS computer programs. The number of disk drives and controllers shall be a function of the computation system design and the total system requirements for on-line real time information flow between the disk(s) and computer(s).

4.1.7.4 Line Printer. A high speed line printer with a minimum speed of 600 lines/minute shall be provided.

4.1.7.5 Magnetic Tape System. A magnetic tape system shall be provided. This system shall provide an on-line, real time backup to the disk system. The magnetic tape system shall be industry compatible and shall meet the following minimum requirements:

(i) Speed: 45 inches per second
(ii) Dual Density: 800/1600 bits per inch
(iii) Number of Channels: Nine (9)

4.1.8 Real Time Clock. A real time clock(s) shall be provided for timing the operation of the complete computation system. The clock(s) shall be settable under program control, readable under program control, and provide interrupts to the processor(s) of the computational system. The resolution of the clock(s)/interval timer shall be at least 500 nanoseconds.

4.1.9 Physical and Environmental Characteristics. The computation system shall operate properly in accordance with the total simulation system requirements and the environmental characteristics of the system. The computation system shall operate with power available at the installation site. Means shall be provided to eliminate or alleviate transients, pulses, or other electrical noise that would cause computer malfunctions. Adequate cooling equipment, including temperature and operating blower sensor protection, shall be provided.

4.1.10 Maintenance Accessibility. The processors and associated equipment shall be constructed in such a manner as to permit free access for trouble shooting and normal servicing. Test points, where applicable, shall be brought to accessible locations. Extender cards shall be provided for each separate type of mechanical card connector utilized.

4.1.11 System Spare Capacity. The contractor shall provide the following spare capacities in the delivered computer system.

4.1.11.1 Control Computer. The control computer of the multiprocessor/multicomputer configuration shall be delivered with the following spare capacity:

a. Time. 50% of the time utilized in the execution of the worst-case cycle of the ADSS program (time-spare = 0.5 (time-used)). The worst-case cycle of the program shall be that cycle which, after execution of all simulation programs or modules, leaves the least amount of time available before the next cycle begins.

b. Memory. 50% of the memory utilized by the program for data and instruction storage (memory-spare = 0.5 (memory-used)). This spare memory shall be comprised of usable blocks, with the minimum contiguous block size being 128 locations.

c. Input/Output Channels. 50% of the input channel capacity used and 50% of the output channel capacity used (I/O-spare = 0.5 (I/O-used)).

d. Mass Storage. 50% of the mass storage capacity utilized by the delivered simulation system shall be provided (disk-spare = 0.5 (disk-used)).

4.1.11.2 DIG General Purpose Computer. The general purpose computer portion of the DIG in the multiprocessor/multicomputer configuration shall be delivered with the following spare capacity.

a. Time. 25% of the time utilized in the execution of the worst-case cycle of the ADSS program (time-spare = 0.25 (time-used)). The worst-case cycle of the program shall be that cycle which, after execution of all simulation programs or modules, leaves the least amount of time available before the next cycle begins.

b. Memory. 25% of the memory utilized by the program for data and instruction storage (memory-spare = 0.25 (memory-used)). This spare memory shall be comprised of usable blocks, with the minimum contiguous block size being 128 locations.

c. Input/Output Channels. 25% of the input channel capacity used and 25% of the output channel capacity used (I/O-spare = 0.25 (I/O-used)).

d. Mass Storage. 25% of the mass storage capacity utilized by the delivered simulation system shall be provided (disk-spare = 0.25 (disk-used)).

4.1.12 System Growth Capacity. The computational system shall be designed to permit expansion of computational capacity without significant design changes to existing hardware and without obsoleting the existing equipment. The following expansion capability shall be provided.

4.1.12.1 Directly Addressable Memory. 100% expansion of the total directly addressable computer system memory. This requirement shall be satisfied by providing the capability to increment memory size by means of the addition of memory modules or, if necessary, to satisfy the processing requirements, the addition of processors with associated memory.

4.1.12.2 Mass Storage. 100% expansion of the on-line mass storage system capacity.

4.1.12.3 Processing Capacity. 100% increase in the maximum computational processing capacity (instructions/second) by the addition of central processors of the same manufacturer's model and series designation as those selected for the initial system.

4.1.12.4 Input/Output. 100% increase in the input/output channels and complete interface associated with the computational system.

4.1.12.5 Interrupt. Must be expandable to 100 levels.

4.2 Computer System Software

Computer software shall be developed to support the total mission of the ADSS including demonstration, testing, and maintenance.

4.2.1 Program Organization and General characteristics. The total computer program system shall be top-down designed and organized into functional modules so that separately identifiable functions can be handled independently to the greatest extent possible. The modules shall load, be modified, and assemble or compile independently. Each module shall be functionally defined, including the development of explicit descriptions of all input/output relationships with equipment and other functional modules. Examples of functional modules include engines, flight equations, ground dynamics, and numerical integration routines. The total program system shall be configuration managed at the module level.

4.2.1.1 Programming Languages. FORTRAN (ANSI X3.9 - 1974) shall be used except for those functional modules satisfying either or both of the following criteria:

a. The module is extremely time-critical or time-consuming, and it can be clearly demonstrated by the contractor that the fidelity of the simulation would suffer and/or the cost of the total system would be significantly increased if it were to be programed in FORTRAN.

b. The programing methods required for the module are ones which are not easily facilitated using FORTRAN because of its constructs. As a result, the object code would clearly lack efficiency if FORTRAN were to be used. For modules satisfying the above criteria, assembly language shall be used.

4.2.1.2 Programing Methods. The following structured programing concepts and techniques shall be applied where-ever possible.

a. Single-entry/single-exit control structures.

b. For function modules coded in FORTRAN, maximum or, ideally, exclusive use of sequences, DO WHILE, and IF THEN ELSE statements or emulations thereof; and minimum use of GO TO statements.

c. Segmentation of code into reasonable amounts of logic that are easily understood.

d. Indentation of program statements within function modules and segments of code to clearly identify logical units and various program levels.

4.2.2 Supervisor Program

A supervisor program shall be designed to maintain control over all real time programs. The supervisor shall operate at the highest iteration rate and shall control all other modules on a periodic basis within the cycle and frame timing structure of the real time system and shall be under the control of the vendor supplied operating system.

4.2.3 Simulation Programs

Programs shall be written to accurately simulate the dynamics of all systems. As a minimum, the following special provisions and programs shall be provided.

4.2.3.1 Target Control. The contractor shall provide programs to maneuver the targets in accordance with a pre-determined profile to create a combat scenario. Ten scenarios shall be written as specified in paragraph 4.2.3.4.

4.2.3.1.1 Record/Playback Demonstrations. The contractor shall provide the necessary methods, software, and hardware to record and/or to playback any test and demonstration in the ADSS. The recording technique shall be such that during playback for demonstration, the visual is activated in accordance with the recorded performance. The recording shall be stored on disk and shall be readily accessible.

4.2.3.1.2 Scoring of Combat Scenarios. The contractor shall provide means by which to record and score combat engagements and ordnance deliveries (e.g., hits, misses) and to provide mission summary feedback to the evaluator within one minute. The contractor shall provide the capability for the operator to optionally choose or delete data recording and scoring output. Weapons delivery scoring will consist of the following for missiles, distance and relative position out of kill envelope (for misses) for hit, indication of a hit is satisfactory.

Up to eight parameters of the simulated aircraft at time of weapon release shall also be recorded, as a minimum. These parameters shall include at least the following: airspeed, altitude, pitch attitude, roll attitude, and g-force. All parameters and scores shall be available on an off-line hard copy printer.

4.2.3.2 Parameter Freeze. Means shall be provided to selectively freeze pitch, roll, altitude, heading, or airspeed. The purpose shall be to facilitate demonstrations and tests.

4.2.3.3 Problem Freeze. Means shall be provided to freeze the total simulation at a prevailing configuration at any time during a simulated mission. The purpose shall be to facilitate demonstrations and tests.

4.2.3.4 Initial Problem Conditions. Programs shall be provided to permit the operator or user to select one of ten sets of initial problem conditions for automatic implementation prior to a simulation run. The ten sets shall be stored in computer memory and shall be selectable from the operating console. The parameters in each set shall be addressable from the operating console to permit assignment and/or changing of values for a particular simulator run or, selectivity, for restorage as a revised set of initial conditions for subsequent recall. The contents of each set of conditions shall be displayed to the operator upon demand. The initial condition parameters shall include those pertaining to basic configuration (e.g., velocity, attitude) and geographical location. Activation of the initial conditions shall not upset operation or calibration of equipment.

4.2.3.5 Numerical Integration. A method of numerical integration and an integration interval, Delta T (ΔT), shall be derived or selected for each part of the problem. The method and associated ΔT shall satisfy stability requirements of the problem, as determined by considering at least the critical Eigenvalues or worst-case natural frequencies of the system to be solved as well as system time delays. The derived or selected integration method and ΔT shall satisfy accuracy requirements of the problem, considering truncation, round off, and propagated errors.

4.2.3.6 Interation and Input/Output Rates. Iteration rates and corresponding input/output rates shall be chosen to assure the following:

a. Model accuracy and stability requirements. The flight program shall cycle at least as fast as the DIG program.

b. Elimination of perceptible time differences between pilot input and simulator response greater than, or less than, that which would occur in the aircraft. The visual and flight cues shall occur within 200 milliseconds of each other.

c. Smooth operation of instruments.

d. That information processed by the real world system and perceivable in the cockpit is not missed by the simulation processing.

e. Compatibility with the visual system. There shall be no noticeable stepping or strobing effect of the visual during high speed rolling, spinning, or yawing of the simulated aircraft.

4.2.4 Systems Programs. Systems programs shall be provided to facilitate preparation, loading, debugging, terminal job entry, test editing, file management, execution, and maintenance of the simulation programs and additional software to be prepared and implemented after system delivery and installation.

4.2.4.1 Operating System. A resident commercially avaiable operating system (OS) shall be provided. The OS shall be resident and operational during loading, debugging, assembling, compiling, and execution of the real time simulation programs. The OS shall provide and control the capability for background processing and concurrent multiple time sliced secondary tasks to be performed during a simulation run. This background processing shall include the compilation and assembly of user programs and all other background processors required to build load modules. The OS shall provide as a minimum the following:

a. Symbiont operation of card reader and line printer
b. File maintenance and security services
c. Communications between all active tasks
d. Terminal job entry services
e. Task roll-in, roll-out capability
f. On-line diagnostic support
g. Automatic error logging
h. Memory management for hardware map

4.2.4.2 Compiler. A FORTRAN (ANSI X3.9 - 1974) compiler shall be provided.

4.2.4.3 Assembler. The standard assembler programs available from the computer manufacturer shall be provided.

4.2.4.4 Loaders. The standard loader programs available from the computer manufacturer shall be provided.

4.2.4.5 Peripheral Operating/Handler Systems. Computer program operating/handler systems shall be provided for all peripherals delivered with the simulator.

4.2.4.6 Mathematical Library. The complete library of standard computer manufacturer mathematical routines shall be provided. In addition, all non real-time/real-time mathematical subroutines used in the simulator computer programs shall be included in this library.

4.2.5 Utility Programs

Utility programs shall be provided to facilitate program debug, maintenance, and input/output operations.

4.2.5.1 Memory Dump. These programs shall be capable of dumping the contents of any number of specified memory locations and special registers. The program shall be able to dump onto the peripheral devices provided with the computation system including magnetic disk, magnetic tape, typewriter, and line printer. Dump formats shall include at least octal or hexadecimal (depending on computer), mnemonic, and the form required by the loader programs.

4.2.5.2 Copy. Utility programs shall be provided to facilitate disk-to-disk copying of data in object code as well as source code format.

4.2.5.3 Input/Output. Programs shall be provided to accomplish input from and/or output to all standard and nonstandard peripherals. This shall include all computation system peripherals, interface equipment, and the operator station.

4.2.6 Verification Programs

Verification programs shall be developed to verify the accuracy of the real-time simulation programs and to verify the operational cycle time and spare time remaining within the cycle and frame periods. These verification programs shall be designed to run within the delivered computer configuration.

4.2.6.1 Cycle Time. A program shall be prepared and delivered to determine the time actually required by each central processor to execute the simulation program or selected parts of a program. This measurement shall provide as a minimum the following items of data.

a. The average program execution time (cycle time).
b. The worst-case execution time (cycle time) encountered in all logically possible paths through the program. (This program will be used in determining spare computing time available to meet the spare time requirements for the simulator system.)
c. Average frame execution times.
d. Frame identity of the longest frame time over a test run and corresponding frame execution time.

4.2.7 Maintenance and Test Programs. Programs shall be provided to fully test the operation of the computer system and the other ADSS equipment. When a malfunction occurs in the computer system or other equipment, these programs shall provide sufficient information to the operator to facilitate location and correction of the malfunction. They shall be capable of running with a minimum of operator intervention.

4.2.7.1 Daily Readiness Check. A daily readiness check program shall be provided. This program shall enable operating personnel to easily determine if the ADSS is ready for operation. The check shall utilize automatic sequencing through a series of standard static outputs utilizing the normal iteration rate of the simulation program. These tests shall enable the operator to ascertain visually that the ADSS is performing properly. Provision shall be incorporated to step through the program incrementally to verify the desired output at each step. The operator shall have the option of either proceeding after he has noted discrepancies or stopping the computer system to perform maintenance. The programs shall not require more than 20 minutes (10 minutes desired) running time.

4.2.7.2 Accuracy and Calibration. Accuracy and calibration test programs shall be provided to check the accuracy and flow of signals, both statically and dynamically through the full range of variables, between the computer and all signal sources and terminal points. This program shall also be used to calibrate the interface equipment, displays and controls, and to determine whether the inputs provided by the simulation program are being transmitted properly to the ultimate destination.

4.2.7.3 Computer Diagnostics. A complete set of diagnostic programs shall be provided to isolate failures in the digital computer. All diagnostic programs shall be fully automatic to the extent that once the operator has loaded the program and set up initial conditions, the program will automatically generate error indications. Commercially available programs shall be used to satisfy these requirements. Any variation or deviation from these requirements shall be explained and justified. These diagnostics shall check the main-frame and main-frame options, and all memory units, peripheral units, and arithmetic units. In addition, diagnostics shall be provided to check the input/output handlers and units. On line main frame diagnostics shall run under control of the OS.

4.2.7.4 Real-Time Interface Equipment Diagnostics. Programs shall be provided that will enable on-line program control checkout of the simulator interface equipment. They shall be of an automated type requiring a minimum of technician effort and shall provide a hard copy of the test results and identify the malfunctions to the card level as a minimum. These programs shall perform tasks specified herein.

4.2.8 Other Software

All other vendor-supplied software which is normally delivered as a part of the purchased computer system shall be provided with the simulation system in machine readable sources.